

UNIVERSIDAD CARLOS III DE MADRID

ESCUELA POLITÉCNICA SUPERIOR

GRADO EN INGENIERÍA DE SISTEMAS AUDIOVISUALES



TRABAJO FIN DE GRADO

*CLASIFICACIÓN DE IMÁGENES DE TRÁFICO PARA UN SISTEMA
INTELIGENTE DE TRANSPORTE*

AUTORA: ESTER BEJERANO MORALES

TUTOR: MARCELINO LÁZARO

JULIO DE 2012

Trabajo Fin de Grado
CLASIFICACIÓN DE IMÁGENES DE TRÁFICO PARA UN
SISTEMA INTELIGENTE DE TRANSPORTE

Autora
ESTER BEJERANO MORALES

Tutor
MARCELINO LÁZARO

La presentación del presente Trabajo Fin de Grado se realizó en julio de 2012, siendo calificada por el siguiente tribunal:

PRESIDENTE: MARÍA CARMEN CALDERÓN PASTOR

SECRETARIO: VICENTE LUQUE CENTENO

VOCAL: VÍCTOR PEDRO GIL GIMÉNEZ

y habiendo obtenido la siguiente calificación:

CALIFICACIÓN:

Leganés, julio de 2012

A los que cada día permanecen a mi lado, con todo mi cariño.
Por muy larga que sea la tormenta, el sol siempre vuelve a brillar entre las nubes.
Khalil Gibran (1883-1931)

Agradecimientos

En primer lugar, quiero agradecer al tutor de este Trabajo Fin de Grado, Marcelino Lázaro, su apoyo y dedicación a lo largo del mismo. Gracias por su paciencia, sus sabios consejos y por brindarme la oportunidad de realizar este trabajo.

Quiero agradecer a las personas que han vivido y sufrido en primera persona esta experiencia, mis padres, M^a José y José, su apoyo incondicional aportado cada día durante esta etapa tan importante de mi vida, sin dejar de lado a mi hermano Víctor que de buena gana me ha asesorado y respaldado en numerosas ocasiones tras su experiencia vivida como estudiante en la Universidad Carlos III. Gracias a mi novio, Javi, por todo el ánimo, afecto y comprensión, a mis abuelos por su cariño y en general a toda mi familia que ha estado a mi lado y que sin ellos esto no hubiese sido posible. Y como no nombrar a la alegría de la casa, mis agapornis Coco y Kitty. Gracias a todos y cada uno por sacarme siempre las mejores de mis sonrisas, aún en los peores momentos de estrés, gracias.

Quiero dar las gracias a una persona muy especial, la señora María que allá por donde iba desprendía una paz increíble y que a día de hoy aunque ya no está presente, nadie ha logrado ni logrará reemplazarla.

Gracias a todos mis compañeros por todos los buenos momentos vividos en la universidad, en especial a Nuria mi querida compañera de prácticas y a todos los profesores que he tenido el placer de conocer y me han impartido clase a lo largo de esta carrera.

Gracias a todos.

Resumen

El presente Trabajo Fin de Grado se encuadra en el ámbito de los Sistemas Inteligentes de Transporte. En la actualidad estos sistemas son de gran interés, ya que gracias a ellos es posible reducir el número de accidentes producidos en carretera, prevenirlos, mejorar la seguridad vial o evitar la congestión de tráfico.

En concreto, este trabajo se basa en diseñar un clasificador para la detección de patrones en imágenes con aplicación a la clasificación de señales de tráfico. El clasificador está formado por 4 módulos funcionales, el módulo de preprocesado empleado normalmente para dar solución a posibles imperfecciones presentes en las imágenes, el módulo de normalización cuyo objetivo es mejorar el funcionamiento del sistema restringiendo el rango dinámico de las señales de entrada, el módulo de extracción de características encargado de obtener descripciones representativas de las imágenes que faciliten el reconocimiento de un objeto concreto presente en las mismas y el módulo máquina encargado de clasificar a qué clase, de entre todas las posibles, pertenece una determinada imagen a partir de sus características.

En el trabajo, para cada uno de estos cuatro módulos se analizarán las prestaciones, para la aplicación propuesta en el mismo, de distintas técnicas. De este modo se realizarán diferentes pruebas empleando configuraciones distintas del clasificador. Se evaluarán por un lado diversas técnicas de preprocesado y se realizarán estudios comparativos entre distintas técnicas de extracción de características, tanto supervisadas como no supervisadas, empleando para la clasificación dos de los métodos de aprendizaje máquina más habituales, las máquinas de vectores soporte y las redes neuronales. Se estudiarán por tanto cuáles son las prestaciones obtenidas según la configuración empleada, y se seleccionarán aquellas que proporcionen resultados adecuados o lo suficientemente elevados, considerando también en la elección la complejidad de la configuración. El objetivo será obtener la máxima precisión posible a la hora de clasificar a qué clase pertenece cada una de las señales de tráfico, teniendo en cuenta las restricciones de implementación de este tipo de clasificadores en aplicaciones en el ámbito de los Sistemas Inteligentes de Transporte.

Índice General

1. Motivación y objetivos	1
1.1. Motivación del trabajo	1
1.2. Objetivos	7
2. Estado del arte y planteamiento del problema	9
2.1. Clasificación de patrones en imágenes	9
2.2. Notación utilizada	10
2.3. Preprocesado de los datos	10
2.4. Normalización	12
2.4.1. Normalización de 0 a 1	12
2.4.2. Normalización de -1 a 1	12
2.4.3. Normalización por varianza	12
2.5. Extracción de características	12
2.5.1. Métodos lineales y métodos no lineales	13
2.5.2. Métodos supervisados y métodos no supervisados	14
2.5.3. Métodos más habituales de extracción de características	14
Análisis de componentes principales (PCA)	14
Análisis de componentes independientes (ICA)	15
Mínimos cuadrados (LS)	16
Análisis lineal discriminante (LDA)	17
2.6. Clasificadores máquina	17
2.6.1. Redes neuronales	17
Perceptrón Multicapa	18
Red de Funciones de Base Radial (RBF)	18
2.6.2. Máquinas de vectores soporte (SVM)	20
Caso separable linealmente	20
Caso no separable linealmente	24
SVM no lineal	25
2.6.3. Comparación entre las dos alternativas	27
2.7. Entrenamiento de la máquina	28
2.7.1. Problema de sobreajuste y subajuste	28
2.7.2. Elección función núcleo (SVM) y función de activación (GRBF)	29
2.7.3. Entrenamiento y evaluación de la máquina	29
2.7.4. Coste computacional	30
2.8. Clasificación multiclase	30
2.8.1. Uno contra todos	30

2.8.2. Todos contra todos	30
3. Requisitos y diseño de la solución	33
3.1. Requisitos y especificaciones para la solución	33
3.2. Esquema básico	34
3.3. Preprocesado	34
3.4. Extracción de características	34
3.5. Normalización	35
3.6. Clasificador máquina	35
4. Resultados y evaluación	37
4.1. Generación de la base de datos para el estudio	37
4.2. Librerías empleadas	39
4.3. Resultados sin ruido	40
4.3.1. Resolución de imágenes	40
4.3.2. Métodos de normalización	43
4.3.3. Número de características	43
4.3.4. Comparación de técnicas de extracción para SVM y GRBF	44
Número de características óptimo	44
Número de neuronas óptimo	47
Comparación de técnicas de extracción de características	49
4.3.5. SVM vs GRBF	52
4.4. Resultados con ruido	53
4.4.1. Entrenamiento sin ruido	53
Comparación de técnicas de extracción de características	53
4.4.2. Entrenamiento con ruido	54
Comparación de técnicas extracción de características	54
4.4.3. Resolución de imágenes	56
4.4.4. Métodos de normalización	57
4.4.5. Comparación de técnicas de extracción para SVM y GRBF	57
Número de características óptimo	58
Número de neuronas óptimo	60
Comparación de técnicas de extracción de características	61
4.4.6. Técnicas de preprocesado	65
4.4.7. SVM vs GRBF	66
4.5. Bloque de extracción de características	66
4.5.1. Extracción de características en general SVM	67
4.5.2. Extracción de características en general GRBF	67
5. Presupuesto y planificación del trabajo	69
5.1. Planificación y seguimiento del trabajo	69
5.2. Presupuesto	71
6. Conclusiones	73
A. Glosario de acrónimos	77

Capítulo 1

Motivación y objetivos

En este primer capítulo se va a motivar el trabajo, haciendo referencia al marco socio-económico en que se ubica, y a la normativa de aplicación general en dicho marco, y se describirán los principales objetivos de este Trabajo Fin de Grado.

1.1. Motivación del trabajo

En el presente trabajo se trata de resolver un problema relacionado con el perfil de la titulación, de interés actual, como es la clasificación de patrones en imágenes, aplicada en el marco de los Sistemas Inteligentes de Transporte. Se entiende por Sistema Inteligente de Transporte o ITS (*Intelligent Transport System*) un sistema basado en un conjunto de soluciones tecnológicas que tratan de solventar los diferentes problemas ocasionados por el aumento de desplazamientos de los vehículos en el sistema de transporte considerado. Este tipo de sistemas se han ido desarrollando como consecuencia de la evolución de la informática, electrónica y la comunicación, y aplican por tanto los avances en las conocidas como Tecnologías de la Información y las Comunicaciones (TICs) [[Unión Europea, 2012](#)]. Debido a que algunas de las soluciones utilizadas están relacionadas con la Telemática, en ocasiones estos sistemas responden a las siglas ATT (*Advanced Transport Telematics*). Los ITS se han ido desarrollando para múltiples medios de transporte: marítimo, aéreo, ferrocarril, por carretera, etc. Este trabajo en concreto, se centrará en el medio destinado al transporte en carretera, puesto que es el medio más común y empleado en la actualidad por un mayor número de usuarios.

En el ámbito del transporte por carretera estos sistemas son de gran interés, ya que gracias a ellos es posible disminuir el número de accidentes producidos en carretera, mejorar la seguridad vial, reducir la congestión o el consumo energético, e incluso reducir la contaminación en las grandes ciudades, que se debe en gran parte a las emisiones de los automóviles (ver Figura [1.1](#)).

La tecnología que emplean los ITS se desarrolla con rapidez. A medida que la tecnología avanza, las posibilidades para incluir aplicaciones y funcionalidades en los sistemas aumentan y los resultados son cada vez más completos y precisos. Para cumplir los objetivos básicos de un ITS es necesario desarrollar múltiples sub-sistemas, cada uno con



Figura 1.1: Efecto visible de la contaminación en una gran ciudad, que en gran medida es producida por las emisiones de automóviles.

su función específica, que se integren en un sistema global de transporte inteligente por carretera. Sistemas como el control de crucero, los sistemas de detección automática de matrículas o señales de tráfico, los sistemas de gestión de aparcamiento, el sistema de ayuda a la frenada de emergencia, o el piloto automático, son solo algunos ejemplos de las funcionalidades requeridas. Además, los ITS permitirán la comunicación en tiempo real entre vehículo-infraestructura (V2I: *vehicle-to-infrastructure communication*) o vehículo-vehículo (V2V: *vehicle-to-vehicle communication*), intercambiando información que facilitará la gestión del tráfico y la comodidad de los usuarios, ya sea en transporte público o privado. Los distintos sub-sistemas que se van a integrar en un ITS se encuentran en la actualidad en distintas fases de investigación y desarrollo y son sin duda la base para el futuro del transporte en carretera, ya que facilitan enormemente la conducción o el control en diversas situaciones. Para su implementación, existe aún un amplio abanico de vías de estudio abiertas y que o bien están en desarrollo o bien no tardarán en desarrollarse.

El desarrollo e implementación de distintas soluciones y funcionalidades para los Sistemas de Transporte Inteligente requieren una gran inversión en este marco por parte de las administraciones y de las empresas privadas. Existen diferentes iniciativas a nivel privado llevadas a cabo por distintas marcas de automóvil para desarrollar estos sistemas [BMW España, 2012], [Volvo Cars, 2012], [Diario ABC, 2012b], [Diario Motor, 2008]. Y también las administraciones están realizando una importante inversión para mejorar las infraestructuras con la introducción de este tipo de soluciones. Por ejemplo, en Estados Unidos el Plan Estratégico de Investigación en ITS (*ITS Research Strategic Plan*) prevé la inversión para el período de 2010-2014 de una cantidad de 100 millones de dólares anuales para investigación y transferencia tecnológica. Para el año 2010, algunas de las cantidades asignadas para distintos conceptos fueron las siguientes: hasta 11,5 millones de dólares en comunicaciones vehículo-vehículo, \$9,3 millones en comunicaciones vehículo-infraestructura o \$1,995 millones en la captura y gestión de datos en tiempo real [U.S. Department of Transportation, 2012]. En Finlandia el presupuesto para carreteras públicas en el año 2001 fue de 778 millones de euros [Instituto Español de Comercio Exterior, 2012]. Por otro lado, existe un programa de cá-

maras de seguridad en el Reino Unido que requiere una inversión de 96 millones de libras. En Francia, hay medidas para evitar las colisiones con árboles que requieren la inversión de 1 millón de euros y existe también una medida para el control automático de la velocidad basada en radares donde el despliegue de los mismos supone un coste de aproximadamente 100 millones de euros [Comisión Europea, 2007].

En general la implantación de los ITS en los distintos estados se ha ido haciendo de forma fragmentaria y poco coordinada, sin continuidad geográfica de los servicios proporcionados, que se restringen al ámbito nacional o incluso regional. A nivel europeo, para intentar resolver este problema el Parlamento Europeo y el Consejo adoptaron el 7 de julio de 2010 la Directiva 2010/40/UE [Parlamento Europeo y Comisión, 2010]. Esta directiva establece el marco para la implantación de los sistemas de transporte inteligentes en el sector del transporte por carretera y para las interfaces con otros modos de transporte. El objetivo es contar con una normativa común que asegure una implantación coordinada y eficaz en su conjunto de las soluciones que puedan implementarse en el sector del transporte por carretera en todo el territorio de la Unión Europea. Uno de los aspectos críticos es la definición de las especificaciones necesarias para garantizar la compatibilidad, interoperabilidad y continuidad de la implantación. En la directiva se especifica como objetivo para la Comisión adoptar antes del 27 de febrero de 2013 las especificaciones a través de las cuales se establezcan los requisitos, procedimientos o cualquier otra exigencia necesaria para proceder a la implantación y uso de los ITS. A nivel nacional, el recientemente aprobado Real Decreto 662/2012 de 13 de abril incorpora la Directiva 2010/40/UE al ordenamiento interno español, estableciendo el marco normativo general para la implantación y uso coordinados y coherentes de los ITS en España [Ministerio de la Presidencia, 2012].

Aunque como se ha dicho los ITS están en fase de desarrollo, muchas de las funciones de estos sistemas son ya una realidad, y nuevas funcionalidades se van presentando día a día, algunas de las cuales se incluyen en el marco normativo de forma progresiva. Por citar un ejemplo, a partir de 2013 la Unión Europea hará obligatoria la instalación de serie de un sistema de frenado automático de emergencia en camiones y autobuses de nueva fabricación [Diario Motor, 2011]. Además, existen ya algunos desarrollos pioneros de sistemas de piloto automático que permiten, a distintos niveles, la circulación de vehículos sin conductor. Por ejemplo, el estado de Nevada es el primero en expedir la licencia para que los vehículos circulen solos sin un conductor [El Mundo, 2012], y ya se han realizado pruebas con un vehículo fabricado por Google [Wikipedia, 2012] (ver Figura 1.2). También otros fabricantes, como Volvo [Diario ABC, 2011] o BMW [Diario ABC, 2012a], [BMW España, 2012] han realizado pruebas recientemente de sistemas de piloto automático en sus vehículos, de forma autónoma, o formando parte de los denominados *Trenes de Carretera*, donde una caravana de coches conectados únicamente por sus sistemas electrónicos siguen a un vehículo *locomotora* que sería el único con conductor [Diario ABC, 2012b] (ver Figura 1.3).

A nivel de componentes, un sistema ITS está formado por un conjunto de elementos hardware y software que emplean tecnología actual. Los principales componentes se podrían clasificar, teniendo en cuenta su función, en los siguientes tipos: sensores, sistemas de comunicaciones, sistemas de procesado, y actuadores.

Los sensores son dispositivos que detectan una acción, externa o interna, y la transmiten adecuadamente. Cada vez es más común ver a los automóviles dotados de sensores



Figura 1.2: Vehículo del programa *Google Driverless Car* (fotografía de la agencia REUTERS) y conductor del mismo en una prueba de conducción en el estado de Nevada (You Tube).



Figura 1.3: Tres coches del fabricante Volvo siguen a un camión en un tren de carretera y siguen la misma ruta que este sin necesidad de conductor (fotografía de la agencia EFE publicada en el diario ABC).

distribuidos en distintos lugares del coche no sólo para monitorizar sus movimientos sino también, para controlar la conducta de la persona que está a cargo del vehículo. Por ejemplo, existen sensores para obtener la velocidad, aceleración y frenado de los vehículos, la distancia de seguridad, e incluso sensores que detectan cuando se produce un accidente para permitir disparar los airbags como medida de seguridad.

Desde el punto de vista de la comunicación, un sistema inteligente está basado principalmente en redes vehiculares, redes VANET (*Vehicular Ad Hoc Network*). Estas son redes ad-hoc móviles que emplean el medio inalámbrico para comunicarse. Poseen las mismas limitaciones que las comunicaciones inalámbricas en cuanto al rango de transmisión. Por este motivo, emplean un modo de funcionamiento multisalto (ver Figura 1.4 donde se ilustra cómo sería el funcionamiento de dicho modo si un vehículo se saliese de la carretera), donde el mensaje emisor es encaminado a través de otros vehículos próximos mediante saltos hasta alcanzar el vehículo receptor. Para obtener más información consultar: [Rodríguez García, 2009], [Hartenstein y Laberteaux, 2010], [Bernardos et al., 2007].

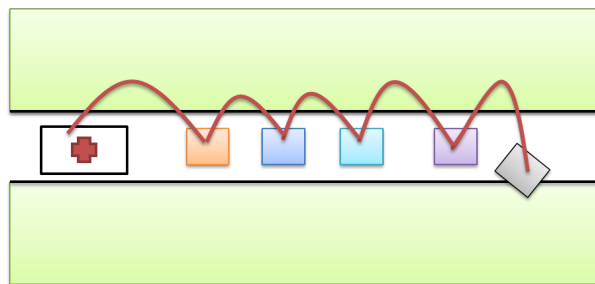


Figura 1.4: Modo de funcionamiento multihop.

Estas redes persiguen el intercambio de mensajes con otros vehículos (V2V) e infraestructura de red de la carretera (V2R: *vehicle-to-roadside communication*) [Arndt, 2008], [Motsinger y Hubing, 2007]. En [Galaviz Mosqueda et al., 2011] se puede consultar la taxonomía de los sistemas de comunicación vehiculares.

El procesado de información es un elemento fundamental para un ITS. Es necesario realizar un procesado de señal en cuanto a comunicaciones, y también en lo que se refiere a la información proporcionada por los distintos sensores para dar una cierta funcionalidad a partir de la información proveniente de los mismos. Toda información recibida debe ser procesada, ya sea debida al intercambio de mensajes entre vehículos, a sistemas de posicionamiento global (GPS), a las medidas de los sensores de a bordo o a la información proporcionada por las infraestructuras situadas en la carretera como pueden ser balizas.

Finalmente, los actuadores son los elementos encargados de tomar acciones, tales como avisos, frenada o control de la velocidad a partir de la información obtenida por el sistema de procesado.

Dentro del procesado de información, un problema que es necesario tratar en varias aplicaciones es el reconocimiento de patrones en imágenes. Es necesario dar solución a este problema por ejemplo para la detección de señales de tráfico, o la identificación de vehículos mediante reconocimiento de sus matrículas, por citar sólo un par de ejemplos. Este Trabajo Fin de Grado se centrará en el reconocimiento de patrones en imágenes con aplicación a los Sistemas Inteligentes de Transporte. En concreto, se va a llevar a cabo el diseño de clasificadores en imágenes aplicado al reconocimiento de señales de

tráfico [Maldonado-Bascón et al., 2007], ya que éstas son de gran importancia de cara a la señalización en carretera para alertar de peligros, proporcionar información o regular el tráfico. Hoy en día algunas marcas, como Opel [Diario Motor, 2008] o BMW [BMW España, 2012], ofrecen ya un sistema de detección de imágenes basado en cámaras, que reconoce señales de tráfico y controla a una cierta distancia de qué tipo de señal se trata indicándoselo al conductor por medio de una pantalla, o proyectándolas sobre el parabrisas (ver Figura 1.5). La cámara suele estar situada en el parabrisas junto al retrovisor interior del vehículo.



Figura 1.5: Esquema con el monitor del Opel Insignia para la monitorización del límite de velocidad (imagen superior) y fotografía con la proyección sobre el parabrisas de la información de limitación de velocidad, prohibición de adelantar, e información del navegador en un vehículo del fabricante BMW (imagen inferior). Ambas imágenes se han obtenido de la página del correspondiente fabricante.

En cualquier caso, la detección de señales de tráfico no sólo es útil para proporcionar ayudas en la conducción, sino que también se están investigando otras posibles aplicaciones como el mantenimiento de señales en autopistas o el inventario de señales en ciudades [Collado et al., 2003]. Además, existen otras aplicaciones que emplean un sistema de detección de patrones en imágenes como son el sistema anti-colisión [Cisneros, 2009b], que se encarga de avisar al conductor cuando detecta la presencia de un objeto, el sistema de aviso de fatiga [Navarrete et al., 2004], que se encarga de advertir al conductor cuando los movimientos del vehículo son sospechosos, el sistema de detección de ángulos muertos [Cisneros, 2009a], que alerta al conductor de la presencia de coches que realizan un adelantamiento o sistemas de detección de peatones [Cisneros, 2011], [Volvo Cars, 2012]. Las técnicas utilizadas para la detección de estos eventos son en muchos casos similares a las necesarias para la clasificación de señales de tráfico.

1.2. Objetivos

Este trabajo se centra en un módulo pequeño pero importante de los Sistemas Inteligentes de Transporte como es la detección de patrones en imágenes y su aplicación a la clasificación de señales de tráfico. La elección del módulo a desarrollar se hizo, por un lado, considerando las limitaciones temporales impuestas en un trabajo como este, y por otro lado, la influencia tecnológica que está suponiendo a día de hoy en el mercado automovilístico. Este módulo estaría integrado con otros subsistemas, como el de segmentación de imágenes [Gonzalez et al., 2009], cuyo proceso consiste en dividir la imagen en regiones u objetos con características similares, y con los distintos actuadores que se utilizarán para monitorizar señales o para tomar medidas a partir de la detección de las mismas (frenado si se excede la máxima velocidad permitida y así se ha requerido previamente por el conductor, por ejemplo). En este trabajo, por las limitaciones propias de un Trabajo Fin de Grado, se considerará el módulo de clasificación de señales de tráfico de forma aislada. Se supondrá que este recibe imágenes ya segmentadas por un módulo previo de segmentación, y se ignorarán las posibles acciones posteriores a la clasificación de una cierta señal.

En el ámbito del problema general de clasificación de señales de tráfico, los objetivos fundamentales de este Trabajo Fin de Grado son los siguientes:

- Diseñar un clasificador de señales de tráfico basado en métodos de aprendizaje máquina, como son las redes neuronales o las máquinas de vectores soporte, con unos requisitos mínimos esperables en un sistema de estas características para su aplicación dentro de un sistema inteligente de transporte.
- Evaluar las prestaciones, en cuanto a probabilidades de error de clasificación, que se pueden obtener con este tipo de clasificadores en la aplicación objeto del Trabajo Fin de Grado.

Además de estos dos objetivos básicos, existen también otros objetivos secundarios que se citan a continuación:

- La generación de una base de datos con imágenes en el ámbito del problema planteado, donde estén presentes algunas de las distorsiones o variaciones que pueden aparecer típicamente en un sistema de estas características. Esta base de datos será utilizada para la evaluación inicial de prestaciones obtenidas con los clasificadores máquina que se diseñen durante el trabajo.
- Comparación de las prestaciones obtenidas para la aplicación de este trabajo de distintas técnicas relacionadas con cada uno de los módulos funcionales necesarios en un clasificador de patrones en imágenes. En particular, se analizarán:
 - Varios métodos de preprocesado y normalización de datos, necesarios para homogeneizar los datos antes de que estos sean presentados al clasificador máquina.
 - Varios tipos de técnicas de extracción de características. La extracción de características es necesaria para reducir la dimensión del espacio de entrada del

clasificador (que en este caso viene dada por el número total de píxeles de la imagen), ya que los métodos máquina aplicados a clasificación sufren de la llamada *maldición de la dimensionalidad*, que básicamente significa que su complejidad aumenta de forma exponencial con la dimensión de los datos a clasificar. Se evaluarán tanto técnicas supervisadas como no supervisadas. Aunque es conocido que las técnicas supervisadas proporcionan en general mejores prestaciones para una complejidad dada, se incluirán en el estudio también técnicas no supervisadas con el objetivo *académico* de evaluar la mejora ofrecida por las técnicas supervisadas frente a las no supervisadas en términos de la relación entre prestaciones y complejidad.

- Varios métodos de aprendizaje máquina como clasificador básico de patrones. En particular, se compararán las máquinas de vectores soporte y las redes de funciones de base radial generalizada, ambas con funciones de activación gaussianas. La comparación se realizará tanto en cuanto a prestaciones (probabilidad de error), como a complejidad (número de parámetros requerido).

A la hora de evaluar todos estos métodos, dadas las limitaciones temporales y de recursos de computación presentes en el desarrollo de un Trabajo Fin de Grado, no será posible realizar la evaluación de todas las posibles combinaciones de técnicas en cada uno de los módulos. Por tanto, se evaluará un subconjunto de las mismas, que se elegirá para que los resultados obtenidos permitan dentro de lo posible obtener información útil de las tendencias que aparecen para esta aplicación con un clasificador basado en métodos de aprendizaje máquina como el que se diseñará.

Capítulo 2

Estado del arte y planteamiento del problema

En este segundo capítulo se explica en qué consiste la clasificación de patrones en imágenes, se introduce la notación empleada, y se presentan algunas de las técnicas de preprocesado de imagen más habituales en el ámbito del trabajo. Se describen además diferentes técnicas de extracción de características y los clasificadores basados en aprendizaje máquina más habituales.

2.1. Clasificación de patrones en imágenes

El problema de clasificación de patrones consiste en resolver a qué clase, de entre un conjunto predefinido de clases posibles, pertenece un cierto patrón (una imagen, por ejemplo, en este trabajo). Una alternativa para llevar a cabo la clasificación, es la utilización de métodos máquina para la obtención de una función de decisión que permita asignar un patrón a su clase correspondiente [Bishop, 2006].

Entre algunas de las aplicaciones típicas del reconocimiento de patrones destacan el reconocimiento de patrones para el diagnóstico en imágenes médicas [Carlot, 2008], el reconocimiento de voz [Merlo et al., 1997], [Milone, 2005] o el reconocimiento de dígitos manuscritos [Seijas, 2003].

La clasificación de imágenes requiere de una serie de módulos como el preprocesado, la normalización, la extracción de características y el módulo máquina. En este capítulo, después de presentar la notación empleada (Sección 2.3) se introducirá la necesidad en este ámbito, de realizar una etapa de preprocesado de la imagen (Sección 2.3), especialmente cuando pueden existir distorsiones, para mejorar su aspecto y facilitar así, el proceso de extracción de características, una etapa de normalización de los datos (Sección 2.4), para mejorar el funcionamiento del algoritmo en cuestión y otra de extracción de características (Sección 2.5) para reducir la dimensión del problema y facilitar en lo posible la capacidad de generalización de los clasificadores propuestos (Sección 2.6).

2.2. Notación utilizada

La notación utilizada en este documento es estándar. Por un lado, las matrices se denotan en mayúscula y en negrita ($\mathbf{X}, \mathbf{Y} \dots$), siendo $\mathbf{X}(a, b)$ el elemento de la fila a y columna b de dicha matriz. Las constantes enteras se denotan en mayúscula y en cursiva ($K, N \dots$), los vectores en minúscula y en negrita ($\mathbf{x}, \mathbf{y} \dots$) y las variables unidimensionales en minúscula y cursiva ($x, y \dots$). Cuando se representan elementos de un conjunto, cada elemento aparece indexado con un subíndice $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$.

Una imagen es habitualmente representada como una matriz con $N_F \times N_C$ píxeles, donde N_F es el número de filas y N_C el número de columnas de la imagen. Si se tiene un conjunto de N_I imágenes, cada una de ellas se puede representar mediante una matriz \mathbf{X}_i de dimensión $N_F \times N_C$, donde $i = 1, \dots, N_I$, y $\mathbf{X}_i(a, b)$ es uno de los píxeles de la imagen. Cada imagen o patrón se puede también representar como un vector fila que se denota como $\mathbf{x}_i = [x_{i,1}, \dots, x_{i,N_D}]$, cuya dimensión será de $1 \times N_D$, donde N_D es el número total de píxeles (dimensión) de la imagen, en este caso $N_D = N_F \times N_C$. En este aspecto, si el número de imágenes empleado es N_I habitualmente se agrupan formando una matriz de dimensión $N_I \times N_D$ donde cada fila se corresponde con una imagen

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \dots \\ \mathbf{x}_{N_I} \end{bmatrix}.$$

2.3. Preprocesado de los datos

Las técnicas de preprocesado se utilizan en muchos casos para tratar de solventar las deficiencias que presentan las imágenes debido a una mala adquisición de las mismas, ya sea por las condiciones de iluminación, por las condiciones atmosféricas o por cualquier otra causa. Entre ellas destacan las técnicas de realce o manipulación del contraste, de suavizado o de detección de bordes [Gonzalez y Woods, 2008], [Gonzalez et al., 2009]. A continuación se describen brevemente algunas de las más habituales.

■ Técnicas de realce

Se aplican a las imágenes cuando la iluminación no es uniforme o simplemente cuando se quiere aumentar el contraste de los objetos presentes en ellas. Emplean para ello recursos como el histograma¹, brillo² y el contraste³ (Figura 2.1).

■ Técnicas de suavizado

¹Histograma, es una función discreta que representa el número de ocurrencias de cada nivel de gris presente en una imagen.

²Brillo, es el valor medio de la imagen, cuyo resultado coincide con el valor medio del histograma.

³Contraste, indica la dispersión de niveles de gris en la imagen. Está asociado a la varianza del histograma.

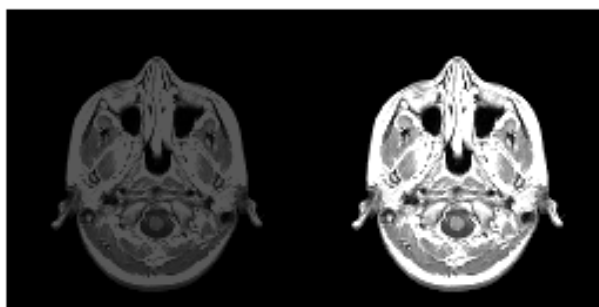


Figura 2.1: Ejemplo de aumento de contraste en una imagen.

Se caracterizan por tratar de eliminar el nivel de ruido presente en la imagen. El ruido se entiende como una variable aleatoria que sigue una función de densidad determinada que puede ser de varios tipos, por ejemplo, gaussiano, impulsional o multiplicativo.

El modelo más común es el ruido gaussiano y se genera en el proceso de captación de la imagen al convertir los fotones en carga eléctrica. El ruido impulsional es conocido también como ruido de sal y pimienta, surge como consecuencia de la diferencia de color o intensidad de los píxeles circundantes. El ruido multiplicativo aparece por falta de iluminación uniforme en la imagen capturada (Figura 2.2).

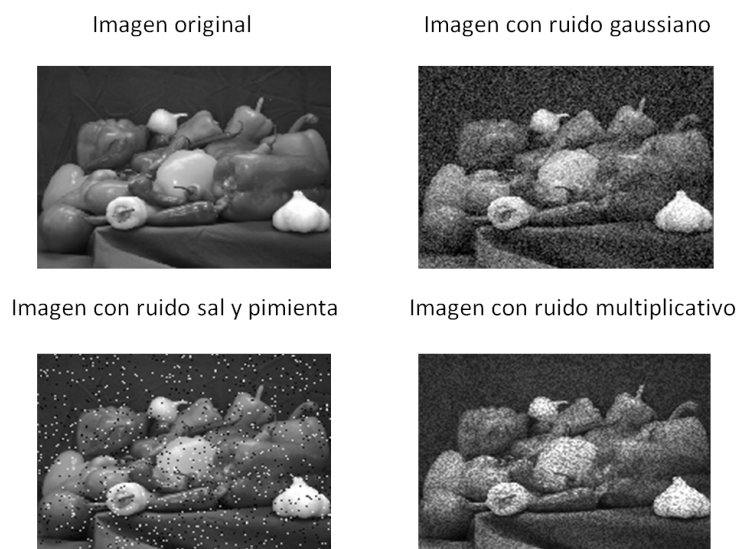


Figura 2.2: Presentación de imágenes con diferentes tipos de ruido.

■ Técnicas de detección de bordes

Se emplean para la extracción de bordes de la imagen para la búsqueda de objetos empleando para ello habitualmente el operador gradiente. Como consecuencia, al aplicar dicho operador en la imagen se localizarán en el módulo del gradiente normalmente, aquellos valores más elevados.

2.4. Normalización

Tras aplicar las técnicas de preprocesado es conveniente llevar a cabo una etapa de normalización. Es habitual en métodos máquina ya que se restringen los valores de tal forma que los datos se encuentren en un rango específico, y facilita con ello la selección de los parámetros de la máquina. En general, dicha etapa produce una mejora en las prestaciones. A continuación, se explican los tipos de normalización más comunes.

2.4.1. Normalización de 0 a 1

Consiste en transformar los datos a una escala acotada que va de 0 a 1. Esto se consigue al dividir cada dato entre el máximo valor alcanzado por ese conjunto

$$\mathbf{x}_n = \frac{\mathbf{x}}{N_{max}}. \quad (2.1)$$

2.4.2. Normalización de -1 a 1

Es el mismo concepto que la técnica presentada en la Sección 2.4.1. Sin embargo, en este caso la escala está comprendida entre -1 y 1, considerando para ello el valor mínimo y máximo que alcanzan los datos

$$\mathbf{x}_n = (1 - (-1)) * \left(\frac{\mathbf{x} - N_{min}}{N_{max} - N_{min}} \right) + (-1). \quad (2.2)$$

2.4.3. Normalización por varianza

Esta normalización persigue que los datos presenten varianza unidad

$$\mathbf{x}_n = \frac{\mathbf{x}}{\sqrt{var}}, \quad (2.3)$$

donde var representa la varianza. Esta normalización se puede aplicar sobre toda la imagen, o sobre cada una de las dimensiones de la misma.

En la literatura se ha visto que no siempre una opción es mejor que la otra, sino que depende de la aplicación. Por ese motivo, se estudiarán por separado cada una de las normalizaciones y se comprobará con ello cuál es la más conveniente para esta aplicación en concreto. Esto se estudiará más adelante en la Sección 4.3.2.

2.5. Extracción de características

La etapa de extracción de características se lleva a cabo antes del proceso de clasificación. Permite obtener descripciones representativas de las distintas imágenes que,

idealmente, deberían facilitar la identificación de la clase a la que pertenece cada imagen. Por ejemplo, en señales de tráfico, un descriptor podría ser la cantidad de un color concreto o la forma de una señal. Dependiendo de las imágenes con las que se trabaje habrá que elegir adecuadamente qué características son las más apropiadas en cada caso y cuáles pueden proporcionar mejores resultados en el proceso de clasificación.

Es importante tener en cuenta que en la clasificación, las características deben de ser lo suficientemente discriminativas como para distinguir objetos de distintas clases. El hecho de emplear un espacio de representación de dimensión elevada puede conllevar a lo que se conoce como la *maldición de la dimensionalidad* [Bellman, 1961] ocasionando problemas en el diseño del clasificador, por lo que si se reduce la dimensión del espacio de características disminuiría con ello la complejidad del clasificador.

El objetivo por tanto, es reducir la dimensión del problema con objeto de facilitar la clasificación de patrones. Una imagen contiene gran cantidad de datos (los valores de cada píxel), sin embargo, muchos de ellos no proporcionan información relevante, o la información que proporcionan es redundante. Por ese motivo, las características extraídas deben contener la máxima información posible, la más relevante, capaz de distinguir unas imágenes de otras. Con ello, se consigue reducir la dimensión de los datos y mejorar así la capacidad de generalización del clasificador.

Existen distintas alternativas para la extracción de características, entre las que destacan la extracción de características heurísticas o la extracción mediante procesado con funciones de coste predefinidas, que es en la que se centra este trabajo. A continuación se presenta una distinción entre las distintas clases de extracción de características atendiendo a dos criterios distintos.

2.5.1. Métodos lineales y métodos no lineales

Se distinguen dos clases de métodos, lineales o no lineales en función de cómo se obtienen las características a partir de los datos.

Un método es lineal cuando la transformación que aplica para obtener las características asociadas a una imagen es lineal. En este caso, la transformación se puede expresar en notación vectorial como una multiplicación

$$\mathbf{x}_i^c = \mathbf{x}_i \cdot \mathbf{A}. \quad (2.4)$$

La matriz \mathbf{A} se corresponde con la matriz de proyección de los datos. Dependiendo del método de extracción de características utilizado, dicha matriz se obtendrá de una manera u otra.

Por otro lado, un método es no lineal cuando la transformación que realiza sobre los datos es una función no lineal

$$\mathbf{x}_i^c = f(\mathbf{x}_i). \quad (2.5)$$

En ambos métodos, el resultado al aplicar la extracción sobre el conjunto de N_I imágenes es una matriz denotada como \mathbf{X}^c , cuya dimensión será de $N_I \times N_c$, donde N_c se

corresponde con el número de características extraídas.

2.5.2. Métodos supervisados y métodos no supervisados

Un método será supervisado o no supervisado, dependiendo de si los datos que se utilizan para realizar la extracción han sido etiquetados previamente o no. Las etiquetas son elementos identificativos de cada imagen, y se denotan como y_i . Permiten por tanto, relacionar a través de ellas a qué clase pertenecen cada una de las imágenes.

Los métodos no supervisados no utilizan información de las etiquetas, es decir, no conocen a priori a qué clase pertenece cada dato (o no emplean dicha información para realizar la operación de extracción). Trabajan únicamente a partir de la información de cómo están distribuidas las muestras sobre el espacio de entrada. Por el contrario, los métodos supervisados, necesitan conocer a priori a qué clase pertenecen los datos y emplean esa información para buscar las características que faciliten separar unas clases de otras de la mejor manera posible según algún criterio previamente establecido.

2.5.3. Métodos más habituales de extracción de características

En esta sección se describen los métodos más habituales de extracción de características indicando de qué tipo es cada uno e incluyendo una breve descripción y referencias bibliográficas para cada método.

Análisis de componentes principales (PCA)

PCA (*Principal Component Analysis*) es un método lineal y no supervisado, y es uno de los más populares para la extracción de características. Un caso particular de la PCA es la transformada de Karhunen-Loève (KLT: *Karhunen-Loève Transformation*).

Consiste en transformar el espacio de representación en uno nuevo, en el que los datos estén linealmente incorrelados. Se trata de encontrar un nuevo conjunto de ejes ortogonales donde la varianza de los datos sea máxima en una dirección determinada, es decir, se basa en situar los nuevos ejes donde hay mayor concentración de energía.

El objetivo es encontrar las direcciones de varianza máxima. Para ello, se dispone de un conjunto de imágenes y de sus correspondientes vectorizaciones $\mathbf{x}_k (k = 1, \dots, N_I)$. Este problema de maximización se resuelve calculando los autovalores y autovectores de la matriz de covarianzas. Si se calcula la media de los patrones de entrada,

$$\bar{\mathbf{x}} = \frac{1}{N_I} \sum_{k=1}^{N_I} \mathbf{x}_k \quad (2.6)$$

la matriz de covarianza se define como

$$\mathbf{C} = \frac{1}{K} \sum_{k=1}^{N_I} (\mathbf{X}_k - \bar{\mathbf{x}}) (\mathbf{X}_k - \bar{\mathbf{x}})^T. \quad (2.7)$$

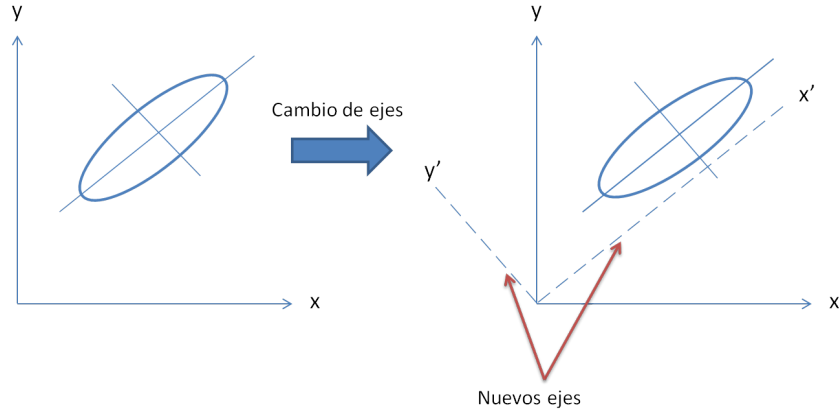


Figura 2.3: Ejemplo de cambio de ejes PCA.

La matriz de covarianza \mathbf{C} tendrá dimensiones $(N_D) \times (N_D)$. Si se realiza una descomposición en autovalores y autovectores, se obtiene

$$\mathbf{\Lambda} = \mathbf{V}^T \mathbf{C} \mathbf{V}, \quad (2.8)$$

donde $\mathbf{\Lambda}$ es la matriz diagonal que contiene los autovalores y \mathbf{V} contiene los autovectores como columnas.

Los autovectores se ordenan por autovalores decrecientes, de tal forma que el autovector con mayor autovalor se corresponde con el eje de máxima varianza, el siguiente autovector con mayor autovalor se correspondería con el segundo eje de mayor varianza y sería ortogonal al anterior y así sucesivamente.

Según el número de características que vayan a representar a la imagen, se escogerán los N_c autovectores de mayor autovalor asociado, siendo N_c la dimensión del vector de características.

Uno de los problemas que presenta PCA es a la hora de trabajar con imágenes donde el tamaño del patrón es elevado. Por ejemplo, si una imagen tiene dimensiones 60×60 píxeles la matriz de covarianzas será muy grande, en concreto de 3600×3600 , por lo que la descomposición en autovalores resulta muy costosa.

Análisis de componentes independientes (ICA)

Es un método lineal y no supervisado. Se trata de una generalización del término PCA. En PCA la incorrelación de las componentes principales no implica su independencia.

El objetivo de ICA (*Independent Component Analysis*) se basa en la técnica conocida como *Blind Source Separation* (BSS) que intenta descomponer las señales de observación como combinación lineal de fuentes independientes.

El modelo ICA se puede representar como:

$$\mathbf{X} = \mathbf{A} \cdot \mathbf{S} \quad (2.9)$$

donde \mathbf{S} es la matriz de variables independientes, \mathbf{X} la matriz de observaciones y \mathbf{A} la matriz de mezcla. Se asume que la matriz de mezcla no es conocida, por lo que se estima a partir de \mathbf{X} . Por otro lado, las variables independientes no pueden observarse directamente, es decir, son variables latentes y se obtendrán también a partir de \mathbf{X} .

Una vez estimada \mathbf{A} , ésta se puede invertir dando lugar a su pseudoinversa \mathbf{W} , conocida como matriz de proyección, y obtener \mathbf{S} como:

$$\mathbf{S} = \mathbf{W} \cdot \mathbf{X}. \quad (2.10)$$

Los algoritmos que implementan ICA se caracterizan por la obtención de la matriz de proyección \mathbf{W} mediante diferentes métodos de optimización. Es importante tener en cuenta que PCA emplea estadísticas de segundo orden ya que trabaja con la matriz de covarianzas; mientras que ICA utiliza estadísticas de orden superior (mayor que dos). Existen múltiples algoritmos desarrollados para ICA. Algunos se basan en emplear el criterio de máxima verosimilitud (ML: *Maximum Likelihood*), maximizar la no gaussianidad (NG: *nongaussianity*) o por ejemplo, minimizar la información mutua (MI: *Mutual Information*) [Hyvärinen y Oja, 2000]. A continuación, se explican dos de los algoritmos más conocidos, ICA-ML y FastICA.

El modelo ICA-ML [Hansen et al., 2001] está ampliamente relacionado con el algoritmo Infomax [Bell y Sejnowski, 1995]. El principio de Infomax está basado en la optimización de redes neuronales, es decir, consiste en maximizar el flujo de salida de una red neuronal con salidas no lineales. La maximización se lleva a cabo a través de un método de optimización basado en gradientes.

El algoritmo FastICA [Hyvärinen y Oja, 2000] es uno de los algoritmos más conocidos, se basa en maximizar la NG. Para la maximización, se emplea un método iterativo de punto fijo en lugar del método basado en gradientes que utilizaba ICA-ML.

Para más información sobre estos métodos consultar: [Hyvärinen y Oja, 2000], [Alvarez y Giraldo, 2008], [Vicente et al., 2007], [Delbracio y Mateu, 2006], [Leiva Murillo, 2007].

Mínimos cuadrados (LS)

LS (*Least Square*) es un método lineal y supervisado. Se emplea para ajustar una función a un conjunto de parejas de datos observadas, es decir, se basa en hallar la función que mejor se ajuste a los datos siguiendo el criterio de mínimo error cuadrático. De esta forma, se persigue que la suma de todos los errores sea mínima. Destaca por su simplicidad computacional.

Ante un sistema sobredeterminado, por ejemplo el sistema de 3 ecuaciones y 2 incógnitas dado en (2.11), surge la necesidad de tener que aproximarlos para poder obtener esa función que mejor se adapte a los datos, en el sentido de tener el mínimo error cuadrático.

$$\left. \begin{aligned} y_0 &= ax_0 + bx_1 \\ y_1 &= cx_0 + dx_1 \\ y_2 &= ex_0 + fx_1 \end{aligned} \right\} \quad (2.11)$$

Dicho sistema se puede expresar en notación matricial como $\mathbf{A} \cdot \mathbf{x} = \mathbf{y}$. La aproximación se basa en obtener \mathbf{x} a través de la pseudoinversa de \mathbf{A} . Para obtener mayor información consultar [Bretscher, 1995].

Aplicado a extracción de características, se puede especificar un cierto valor \mathbf{y} de dimensión N_c a cada clase y buscar la matriz \mathbf{A} que mejor realice la proyección.

Análisis lineal discriminante (LDA)

LDA (*Linear Discriminant Analysis*) es un método lineal y supervisado. Se basa en obtener una proyección de los datos en un espacio de dimensión menor o igual que el de entrada para que la separación entre clases sea la mayor posible. A diferencia de PCA, no busca minimizar el error de representación cometido [Delbracio y Mateu, 2006].

Existen diferentes implementaciones de LDA, entre ellas una de las más conocidas es *Fisher Linear Discriminant Analysis* [Welling, 2005].

2.6. Clasificadores máquina

En esta sección se describen dos de los métodos máquina más habituales para la clasificación de patrones en imágenes, como son las redes neuronales y las máquinas de vectores soporte. En la formulación inicial la descripción se centrará en clasificadores binarios, es decir, clasificadores diseñados para discriminar 2 clases posibles.

2.6.1. Redes neuronales

Surgen en los años 50, y son modelos simplificados de las redes de neuronas que constituyen el cerebro. Una red neuronal forma parte del campo de la inteligencia artificial (IA). Según Simon Haykin [Haykin, 1994], una red neuronal es “un procesador paralelo masivamente distribuido que tiene una facilidad natural para el almacenamiento de conocimiento obtenido de la experiencia para luego hacerlo utilizable”. Se parece al cerebro en dos aspectos:

1. El conocimiento es obtenido por la red a través de un proceso de aprendizaje a partir de ejemplos.
2. Las conexiones interneuronales conocidas como pesos sinápticos son utilizadas para almacenar dicho conocimiento.

Dos de las redes neuronales más conocidas son el Perceptrón Multicapa o MLP (*Multilayer Perceptron*) y la Red de Funciones de Base Radial o RBF (*Radial Basis Function network*) [Park et al., 2002].

Perceptrón Multicapa

Es la red neuronal más común y empleada. Se trata de una red multicapa sin realimentación, es decir, todas las señales se propagan hacia delante a través de las capas. Se caracteriza por tener una capa de entrada, una o varias capas ocultas y una capa de salida. La capa de entrada es la que recibe directamente la información procedente de las fuentes externas a la red, la capa oculta interna a la red donde no existe contacto con el exterior y la de salida es la que envía la información hacia el exterior.

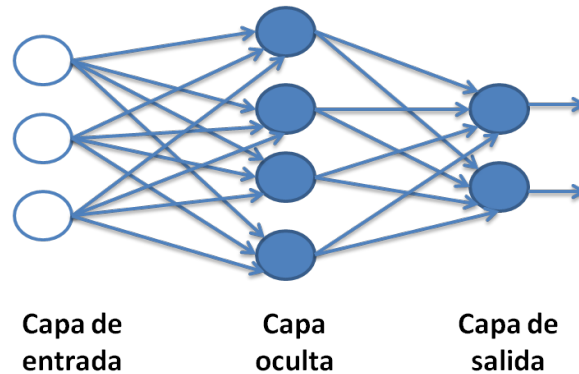


Figura 2.4: Arquitectura típica de una red neuronal hacia delante del Perceptrón Multicapa.

El algoritmo de entrenamiento más frecuente es el algoritmo de retropropagación de errores (back-propagation error) y normalmente se utiliza en problemas de aproximación funcional.

Para más información consultar [Rosenblatt, 1959], [Widrow, 1990].

Red de Funciones de Base Radial (RBF)

Las redes RBF fueron diseñadas como solución al problema de aproximación en espacios multidimensionales. Construyen sus modelos con funciones de activación diferentes para la capa oculta y la de salida. Los primeros trabajos se atribuyen a Powell [Powell, 1985]. Posteriormente, estas funciones fueron empleadas por Broomhead y Lowe en 1988 para el diseño de redes neuronales introduciendo un método alternativo al Perceptrón Multicapa [Broomhead y Lowe, 1988]. Otros estudios importantes se encuentran reflejados en los artículos de Moody y Darken [Moody y Darken, 1990] y Poggio y Girosi [Poggio y Girosi, 1990].

A diferencia del MLP, están formadas por una única capa oculta. Se tiene un conjunto de N datos de entrada y sus correspondientes etiquetas $(\mathbf{x}_k, y_k) \in \mathbb{R}^D \times \{0, 1\}$. La salida de la red RBF para el patrón de entrada \mathbf{x}_k es

$$z_k = z_k(\mathbf{x}_k) = \sum_{i=1}^{N_n} \lambda_i O(\mathbf{x}_k, \theta_i), \quad (2.12)$$

siendo N_n el número de neuronas, λ_i los pesos de las neuronas de salida y $O(\mathbf{x}_k, \theta_i)$ funciones de base radial, pudiéndose escribir como:

$$O(\mathbf{x}_k, \theta_i) = f(\|\mathbf{x}_k - \mathbf{c}_i\|), \quad (2.13)$$

donde θ_i agrupa los parámetros de la i -ésima neurona $\theta_i = [\mathbf{c}_i, \sigma_i] = [c_{i,1}, \dots, c_{i,D}, \sigma_i]$.

Por tanto, como se puede observar en la Ecuación (2.13), la función depende de la distancia al centroide. De ahí el nombre de funciones de base radial. Una de las funciones de activación más empleada es la gaussiana [Haykin, 1994]

$$O(\mathbf{x}_k, \theta_n) = \prod_{d=1}^D \exp \left\{ -\frac{(x_{k,d} - \mu_{n,d})^2}{2\sigma_n^2} \right\}. \quad (2.14)$$

En cuanto al entrenamiento habitualmente se distinguen dos etapas, un proceso de optimización no lineal que consiste en obtener las varianzas y los centroides de las funciones base, y un proceso lineal para estimar los pesos de la capa de salida.

Una manera sencilla de llevar a cabo la optimización no lineal consiste en fijar las varianzas y seleccionar aleatoriamente los centroides o emplear algoritmos de clustering. Cuando las funciones base empleadas son gaussianas una alternativa es utilizar el método OLS (*Orthogonal Least Squares*) [Chen et al., 1991], que calcula los centroides de tal forma que maximicen la varianza.

El principal problema de las RBF es el conocido como la *maldición de la dimensionalidad*. Se produce debido al crecimiento exponencial del volumen asociado al incremento de dimensiones en un espacio matemático, es decir, a medida que aumenta la dimensión del espacio de entrada, el número de neuronas requeridas crece exponencialmente con la dimensión del espacio. Para tratar de solventar este problema, se emplean las denominadas funciones de base radial generalizadas (GRBF: *Generalised Radial Basis Function network*). Son una extensión de las RBF cuya principal diferencia radica en las funciones base. Éstas tienen la posibilidad de presentar una varianza distinta según la dirección del espacio de entrada, abarcando de una forma más precisa los datos. Para ello, se emplea una norma ponderada [Lázaro, 2001]. En ese caso, los parámetros de la neurona son $\theta_i = [\mathbf{c}_i, \sigma_i] = [c_{i,1}, \dots, c_{i,d}, \sigma_{i,1}, \dots, \sigma_{i,d}]$, y su función de activación es

$$O(\mathbf{x}_k, \theta_n) = \prod_{d=1}^D \exp \left\{ -\frac{(x_{k,d} - \mu_{n,d})^2}{2\sigma_{n,d}^2} \right\}. \quad (2.15)$$

Considerando la Ecuación (2.15), los parámetros de la red son los centros (D -dimensional) de cada función base, las varianzas o desviaciones estándar (D -dimensional) y los pesos N_n de la neurona de salida.

$$\mathbf{w} = [\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_{N_n}, \boldsymbol{\sigma}_1, \dots, \boldsymbol{\sigma}_{N_n}, \lambda_1, \dots, \lambda_{N_n}] \quad (2.16)$$

donde

$$\boldsymbol{\mu}_i = [\mu_{i,1}, \dots, \mu_{i,D}], \quad \boldsymbol{\sigma}_i = [\sigma_{i,1}, \dots, \sigma_{i,D}]. \quad (2.17)$$

Los parámetros óptimos se obtienen generalmente minimizando alguna función de coste. La más habitual es la función de error cuadrático medio (MSE: *Mean Squared Error*), que se define como

$$J^{MSE}(\mathbf{w}) = \frac{1}{N} \sum_{k=1}^N (y_k - z_k)^2. \quad (2.18)$$

Dicha función se puede minimizar mediante un algoritmo de descenso de gradiente como sucede en [Santamaría et al., 1999] para una GRBF.

Una vez entrenada la red, la regla de decisión se puede expresar como:

$$\hat{y}_k = \begin{cases} 1, & \text{si } z_k > \frac{1}{2} \\ 0, & \text{si } z_k \leq \frac{1}{2} \end{cases} \quad (2.19)$$

2.6.2. Máquinas de vectores soporte (SVM)

Las máquinas de vectores soporte (SVM: *Support Vector Machines*), conocidas también como máquinas de soporte vectorial, están basadas en la teoría de aprendizaje estadístico desarrollada por Vladimir Vapnik [Vapnik, 1999]. Fue inicialmente presentada como técnica para la clasificación de datos mediante aprendizaje máquina [Schölkopf et al., 1999], [Burges, 1998], aunque posteriormente se extendió también para aplicaciones de regresión. Por otro lado, cabe destacar que los modelos de SVM están ampliamente relacionados con Redes Neuronales [Flórez López y Fernández Fernández, 2008].

El objetivo es entrenar la máquina con una serie de muestras representativas del problema a resolver. Una vez realizado el entrenamiento, la máquina debe ser capaz de clasificar correctamente a qué clase pertenecen nuevas muestras que no han participado en el entrenamiento, para obtener una buena generalización de la misma. Para ello, se define una función de riesgo empírico sobre el conjunto de entrenamiento, y el objetivo del entrenamiento es encontrar los parámetros de la máquina que minimizan dicho riesgo.

A continuación, se presentan los principios básicos de la teoría de las SVM. Se pueden dar diferentes casos dependiendo de si los datos de los que se parte son separables linealmente o no. En concreto, se va a presentar la resolución del problema de clasificación para clasificadores binarios.

Para mayor información consultar [Schölkopf et al., 1999], [González Abril, 2003].

Caso separable linealmente

Se considera que las muestras de las dos clases son linealmente separables, como muestra la Figura 2.5. En ella, se refleja cómo los datos de ambas clases pueden quedar situados a ambos lados de una frontera de clasificación lineal. Pueden existir infinitas rectas que

separen los datos de las dos clases. El objetivo, por tanto, es obtener el hiperplano⁴ más apropiado. En la teoría de la SVM este es el hiperplano de máximo margen que clasifica las muestras sin cometer ningún error, entendiendo como margen la distancia entre los datos y el plano de origen o frontera de decisión.

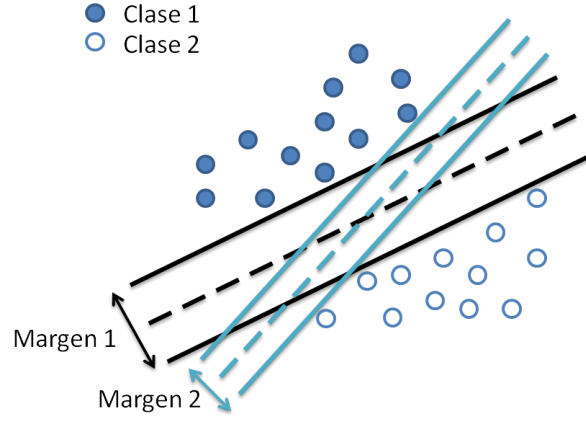


Figura 2.5: Ejemplo de la representación de muestras linealmente separables de dimensión 2 y distintas rectas que pueden separarlas, con sus correspondientes márgenes.

Se parte de unos datos (\mathbf{x}) y unas etiquetas (y) asociadas a éstos que van a indicar a qué clase pertenece cada uno

$$(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_N, y_N) \in \mathbb{R}^D \times \{\pm 1\}, \quad (2.20)$$

donde N es el número de muestras y D el número de dimensiones de los vectores que contienen los datos.

En este caso, cuando los datos son linealmente separables, es fácil encontrar un hiperplano que separe las muestras entre clases, pudiéndose representar como:

$$\langle \mathbf{w}, \mathbf{x} \rangle + b = 0, \quad (2.21)$$

donde el \mathbf{x} representa a las muestras, \mathbf{w} es el vector normal al hiperplano, y el operador \langle, \rangle denota el producto escalar.

La función de decisión correspondiente a la ecuación anterior está definida por la Ecuación (2.22). Ésta determinará a qué clase pertenecen las muestras según \mathbf{x} , de tal forma que cuando la evaluación de la expresión sea mayor que 0 las muestras pertenecerán a una clase y cuando sea menor que 0 a la otra

$$f(\mathbf{x}) = \text{sgn}(\langle \mathbf{w}, \mathbf{x} \rangle + b). \quad (2.22)$$

Interesa obtener los valores de \mathbf{w} y b de tal forma que se obtenga el máximo margen, es decir, la máxima separación entre ambas clases. El clasificador se puede representar

⁴Un hiperplano es la generalización al espacio n-dimensional del concepto de recta.

mediante las siguientes expresiones (ver Figura 2.6):

$$\langle \mathbf{w}, \mathbf{x}_i \rangle + b > +M, \quad \langle \mathbf{w}, \mathbf{x}_i \rangle + b < -M \quad (2.23)$$

para valores $y_i = +1$ e $y_i = -1$, respectivamente, y donde M determina el margen. De forma compacta, estas expresiones se pueden escribir como

$$y_i \cdot (\langle \mathbf{w}, \mathbf{x}_i \rangle + b) \geq M, \quad i = 1, \dots, N. \quad (2.24)$$

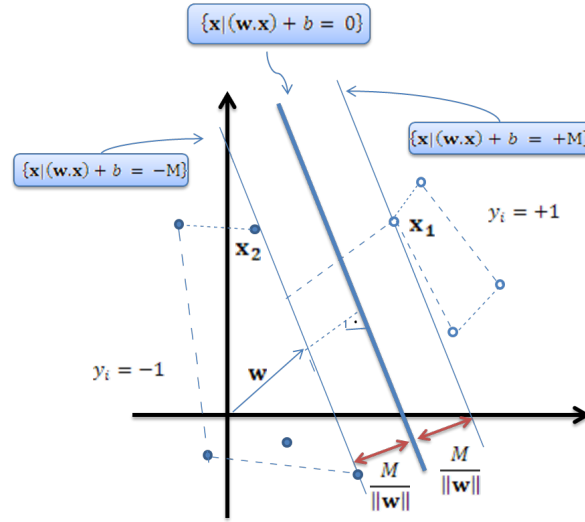


Figura 2.6: Hiperplano de separación óptimo para patrones linealmente separables.

El siguiente paso sería llevar a cabo una normalización, por lo que la ecuación (2.23), quedaría de la siguiente forma:

$$\langle \mathbf{w}, \mathbf{x}_i \rangle + b > +1, \quad \langle \mathbf{w}, \mathbf{x}_i \rangle + b < -1 \quad (2.25)$$

que de forma compacta se puede escribir como:

$$y_i \cdot (\langle \mathbf{w}, \mathbf{x}_i \rangle + b) \geq 1, \quad i = 1, \dots, N. \quad (2.26)$$

En un problema de estas características, habrá un número infinito de hiperplanos que cumplan esa condición. Por ese motivo, se busca aquel que maximice la distancia entre los datos y la frontera de decisión (margen), sin dejar de lado cuál es el verdadero propósito de esta acción, clasificar correctamente nuevos datos una vez entrenada la máquina. En ese caso, si se tienen en cuenta los puntos para los que se cumple la igualdad (2.26) se obtienen dos hiperplanos paralelos entre sí y paralelos a su vez al hiperplano (2.21)

$$\langle \mathbf{w}, \mathbf{x}_i \rangle + b = +1, \quad \langle \mathbf{w}, \mathbf{x}_i \rangle + b = -1. \quad (2.27)$$

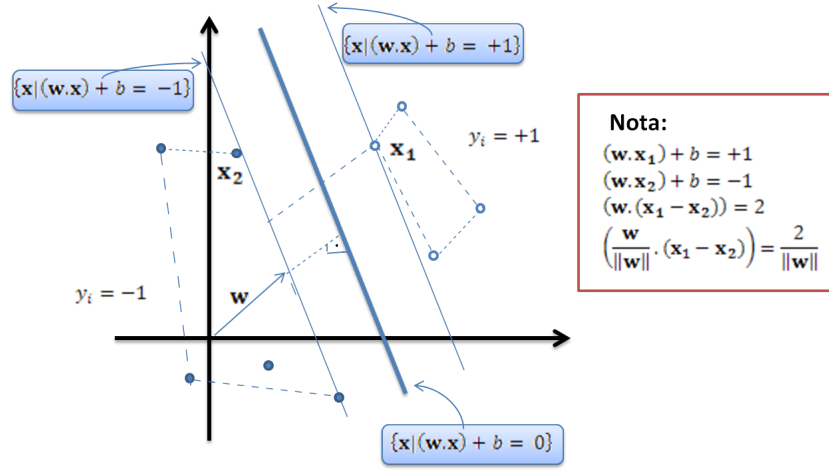


Figura 2.7: Hiperplano de separación óptimo normalizado para patrones linealmente separables.

Teniendo en cuenta la Figura 2.7, la distancia de cada hiperplano al origen es $\frac{1}{\|\mathbf{w}\|}$, por tanto, el margen del hiperplano (2.21) es $\frac{2}{\|\mathbf{w}\|}$. De esta forma, se obtendrá el máximo margen minimizando la siguiente función de coste

$$\tau(\mathbf{w}) = \frac{1}{2} \|\mathbf{w}\|^2, \quad (2.28)$$

considerando las restricciones de (2.26).

La frontera de decisión, va a estar definida por los puntos más próximos al hiperplano de cada conjunto de muestras, como se muestra en la Figura 2.6. A esos puntos que se encuentran justo a la distancia del margen, se les conoce como vectores soporte. En la Figura 2.7, se puede observar que los vectores soporte son aquellos que realmente definen el hiperplano, mientras que el resto son irrelevantes a la hora de definir la frontera.

El empleo de multiplicadores de Lagrange permite tratar este problema de optimización y posibilita la representación del hiperplano como combinación lineal de las propias muestras [Schölkopf et al., 1999]. El lagrangiano viene definido por:

$$L(\mathbf{w}, b, \alpha) = \frac{1}{2} \|\mathbf{w}\|^2 - \sum_{i=1}^N \alpha_i (y_i \cdot (\langle \mathbf{x}_i \cdot \mathbf{w} \rangle + b) - 1) \quad (2.29)$$

donde los multiplicadores de Lagrange se denotan como $\alpha_i \geq 0$. Hay un multiplicador para cada una de las restricciones de (2.26), por tanto, el número de multiplicadores será N . Se quiere minimizar la expresión anterior con respecto a las variables principales (\mathbf{w} y b) y maximizarla con respecto a α_i . Para minimizarla hay que hacer la derivada parcial con respecto a \mathbf{w} y b e igualar a 0:

$$\frac{\partial}{\partial b} L(\mathbf{w}, b, \alpha) = 0 \quad (2.30)$$

$$\frac{\partial}{\partial w} L(\mathbf{w}, b, \alpha) = 0 \quad (2.31)$$

Como consecuencia se obtienen las siguientes expresiones:

$$\sum_{i=1}^N \alpha_i y_i = 0 \quad (2.32)$$

$$\mathbf{w} = \sum_{i=1}^N \alpha_i y_i \mathbf{x}_i. \quad (2.33)$$

Considerando las expresiones anteriores, si el problema planteado es linealmente separable, se puede escribir la función de decisión de la siguiente forma:

$$f(\mathbf{x}) = \text{sgn} \left(\sum_{i=1}^N y_i \alpha_i \cdot \langle \mathbf{x}, \mathbf{x}_i \rangle + b \right) \quad (2.34)$$

No todos los multiplicadores α_i van a ser distintos de 0, sino que únicamente tomarán valor aquellos asociados a las muestras que sean vectores soporte. Por tanto, la Ecuación (2.34) se puede reescribir como:

$$f(\mathbf{x}) = \text{sgn} \left(\sum_{i=1}^{N_{vs}} y_i \alpha_i \cdot \langle \mathbf{x}, \mathbf{x}_i^{vs} \rangle + b \right) \text{ para } \alpha_i \neq 0 \quad (2.35)$$

siendo N_{vs} el número de vectores soporte, y \mathbf{x}_i^{vs} dichos vectores.

Caso no separable linealmente

Hasta ahora, se ha considerado que las muestras eran linealmente separables, es decir, la frontera de decisión se podía definir sin problemas con una recta. Sin embargo, no todos los problemas son linealmente separables, de hecho normalmente suele haber solapamiento entre muestras de las dos clases. Esta vez se persigue encontrar el hiperplano que cometa el menor número de errores posible. Como solución se introducen unas variables de relajación ξ_i en las restricciones cuyo cometido es controlar el error y penalizar las muestras mal clasificadas. Con ello, se relajan las restricciones de la Ecuación (2.26) quedando de la forma:

$$y_i \cdot (\langle \mathbf{w}, \mathbf{x}_i \rangle + b) \geq 1 - \xi_i \quad (2.36)$$

con

$$\xi_i \geq 0. \quad (2.37)$$

Para que una muestra esté mal clasificada, el ξ_i correspondiente tiene que ser mayor que 1. Las muestras bien clasificadas cumplirán $0 < \xi_i < 1$ y su valor dependerá de cómo de cerca o de lejos estén las muestras con respecto a la frontera de decisión.

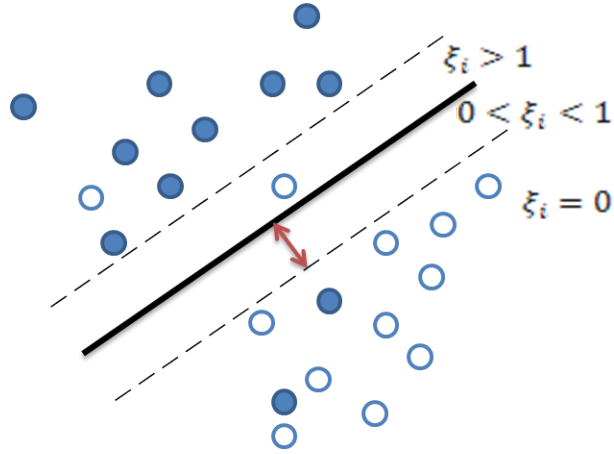


Figura 2.8: Problema no separable linealmente.

Por otro lado, la función de coste a minimizar teniendo en cuenta las nuevas restricciones (2.36) y (2.37) es:

$$\tau(\mathbf{w}) = \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^N \xi_i. \quad (2.38)$$

C es un parámetro ajustable relacionado con la penalización de las muestras mal clasificadas. Dependiendo del valor que tome se dará mayor o menor importancia a minimizar la norma o cumplir las restricciones planteadas. El objetivo es obtener el valor de C que garantice un mínimo error de clasificación para las muestras de entrenamiento y maximice el margen. Si ahora se emplea Lagrange, la Ecuación 2.29 queda:

$$L(\mathbf{w}, b, \alpha, \mu) = \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^N \xi_i - \sum_{i=1}^N \mu_i \xi_i - \sum_{i=1}^N \alpha_i (y_i \cdot (\langle \mathbf{x}_i \cdot \mathbf{w} \rangle + b) - 1 + \xi_i) \quad (2.39)$$

Para conseguir $\xi_i > 0$ se introduce el multiplicador de Lagrange μ_i . A continuación, los pasos que hay que seguir para obtener la solución óptima son los mismos que en el caso separable, es decir, se deriva con respecto a cada una de las variables y se iguala a 0 obteniendo las Ecuaciones (2.32), (2.33) y

$$C - \mu_i - \alpha_i = 0. \quad (2.40)$$

Esto implica que ahora los multiplicadores estarán acotados $0 \leq \alpha_i \leq C$.

SVM no lineal

En muchos casos, la distribución de los datos sobre el espacio de entrada hace que una frontera lineal no sea adecuada para separar las muestras de las dos clases. En este caso, se puede aplicar una transformación al espacio de entrada, $\phi(\mathbf{x})$, para obtener un espacio de características de dimensión superior (espacio Hilbert [Marchena, 2000]) donde aumenta

la posibilidad de que haya separación lineal y por tanto, sea posible trazar el hiperplano óptimo de separación (ver Figura 2.9). La solución en este caso queda en función del producto escalar en el espacio de características en lugar de realizarlo en el espacio de entrada (como ocurría en la Ecuación (2.35))

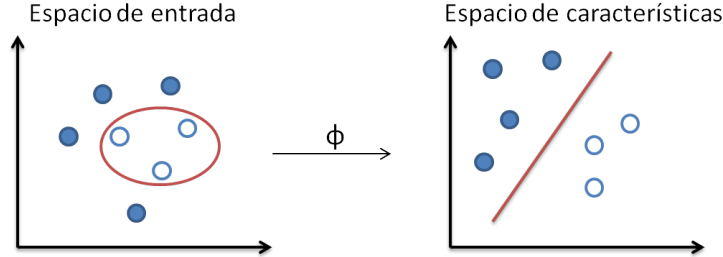


Figura 2.9: Proyección del espacio de entrada al espacio de características.

$$f(\mathbf{x}) = \text{sgn} \left(\sum_{i=1}^{N_{vs}} y_i \alpha_i \cdot \langle \phi(\mathbf{x}), \phi(\mathbf{x}_i^{vs}) \rangle + b \right). \quad (2.41)$$

En muchas ocasiones, no se trabaja directamente con la transformación $\phi(\mathbf{x})$, sino que la proyección del espacio de entrada al espacio de Hilbert se realiza a través de una función núcleo (o *kernel*). La función núcleo se puede definir como:

$$k(\mathbf{x}, \mathbf{x}_i) = \langle \phi(\mathbf{x}), \phi(\mathbf{x}_i) \rangle. \quad (2.42)$$

A este proceso se le conoce como el truco de los núcleos⁵. No es necesario conocer la transformación ϕ , sino únicamente conocer la función núcleo que permite medir distancias de los patrones sobre el espacio de características. La función de decisión queda por tanto:

$$f(\mathbf{x}) = \text{sgn} \left(\sum_{i=1}^{N_{vs}} y_i \alpha_i (k(\mathbf{x}, \mathbf{x}_i^{vs}) + b) \right) \quad (2.43)$$

Toda función $k(\mathbf{x}, \mathbf{x}_i)$ que cumpla el Teorema de Mercer [Schölkopf et al., 1999] es una función núcleo válida. El teorema indica únicamente si un determinado núcleo cumple la propiedad de producto escalar, no proporciona información sobre cómo construir ϕ .

Los núcleos más comunes son:

- **Lineal**

$$k(\mathbf{x}, \mathbf{x}_i) = \mathbf{x} \cdot \mathbf{x}_i \quad (2.44)$$

⁵El truco del núcleo es empleado cuando no se dispone de información de ϕ y por tanto, el cálculo de (2.42) y de la función de decisión sería imposible. Por este motivo, no es necesario conocer ϕ sino únicamente conocer la función núcleo que calcule el producto escalar de los puntos de entrada en el espacio de características. En ese caso, no habrá que preocuparse por la dimensión del espacio y por tanto, el coste computacional se verá reducido.

- **Polinómico**

$$k(\mathbf{x}, \mathbf{x}_i) = (\gamma \mathbf{x} \cdot \mathbf{x}_i + c)^\alpha \quad (2.45)$$

- **Gaussiano**

$$k(\mathbf{x}, \mathbf{x}_i) = \exp(-\gamma |\mathbf{x} - \mathbf{x}_i|^2) \quad (2.46)$$

donde γ es una constante de proporcionalidad, c un coeficiente y α el rango del polinomio.

2.6.3. Comparación entre las dos alternativas

Las máquinas de vectores soporte en la fase de entrenamiento se basan en el principio de Minimización de Riesgo Estructural, mientras que las redes neuronales se basan en la Minimización de Riesgo Empírico. Según algunos autores como Vapnik la Minimización de Riesgo Estructural funciona mejor, ya que se centra en minimizar un límite superior de riesgo esperado a diferencia de minimizar el error sobre los datos de entrenamiento [Vapnik, 1999].

Si se considera el coste computacional que puede conllevar utilizar máquinas de vectores soporte o redes neuronales, es obvio que el entrenamiento es complicado en ambos casos. Sin embargo, una vez obtenidos los parámetros óptimos, el proceso de evaluación es bastante eficaz. Por tanto, se puede decir que es difícil entrenar la máquina y obtener cuáles son los parámetros óptimos, pero una vez conocidos esos parámetros es eficiente evaluar la salida de la máquina para un cierto patrón de entrada. Para ello, sólo hay que evaluar las Ecuaciones (2.12) o (2.43).

Una de las principales desventajas de las SVM es el tiempo de entrenamiento cuando se emplean grandes conjuntos de datos.

Las redes neuronales presentan el problema de mínimos locales. La función de coste, como la Ecuación (2.18), no es convexa (Figura 2.10). Eso hace que sea necesario partir de distintas inicializaciones. Sin embargo, este problema no lo tienen las SVM, ya que el espacio de búsqueda tiene un único mínimo local, es decir, en este caso, la función de coste es convexa (Figura 2.11).

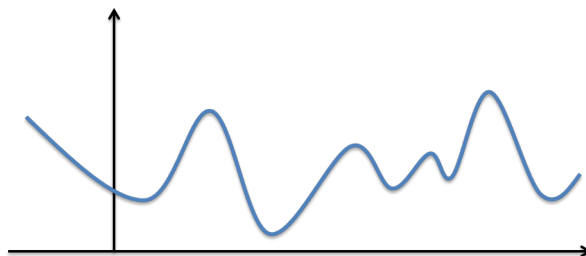


Figura 2.10: Función no convexa con múltiples mínimos locales.

Por otro lado, las redes neuronales presentan la ventaja de poder fijar la complejidad de la solución escogiendo el número de neuronas que se van a emplear. Sin embargo,

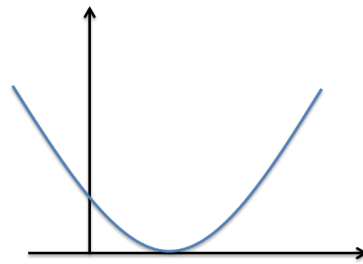


Figura 2.11: Función convexa con un único mínimo local.

en máquinas de vectores soporte la complejidad no se puede fijar, sino que vendrá dada por el número de vectores soporte (aunque es cierto que existen algunas técnicas para reducir a posteriori la complejidad de la solución, como el algoritmo de la pre-imagen [Bakir et al., 2004], [Kwok y Tsang, 2004]. En general son complejas y sensibles al problema de los mínimos locales).

2.7. Entrenamiento de la máquina

En este apartado se hablará de cómo se han escogido los parámetros adecuados de la máquina, que función núcleo (SVM) o de activación de las neuronas (GRBF) se ha empleado y los problemas que pueden surgir a la hora de escoger dichos valores.

2.7.1. Problema de sobreajuste y subajuste

El entrenamiento de la máquina consiste en obtener los parámetros del clasificador, pero para ello debe existir un compromiso entre los parámetros de la máquina, ya que dependiendo de su valor puede dar lugar a problemas como el sobreajuste o el subajuste.

El sobreajuste se produce cuando los parámetros obtenidos en el entrenamiento se ajustan demasiado a las muestras, es decir, se entrena demasiado la máquina. De esa forma cuando se introduce una nueva muestra, la máquina no es capaz de generalizar lo cual provoca una mala clasificación de los datos. El subajuste es el efecto contrario, es decir, no se produce una buena generalización ya que los parámetros se ajustan poco a las muestras. Las Figuras 2.12 y 2.13 ilustran ambos sucesos.

En una SVM, si el número de vectores soporte obtenido para un determinado valor es elevado, normalmente suele derivar en un problema de sobreajuste. Por regla general, cuanto menos vectores soporte se empleen, la solución obtenida va a ser más suave. Por otro lado, cuanto mayor sea el valor de sigma, más ancha será la gaussiana lo que conllevará a un menor número de vectores soporte.

En una GRBF se puede escoger la complejidad de la solución, de tal forma que si el número de neuronas empleado es muy elevado puede producirse sobreajuste. Por el contrario, si el número de neuronas es demasiado bajo podría ocasionar un problema de subajuste.



Figura 2.12: Problema sobreajuste.



Figura 2.13: Problema subajuste.

2.7.2. Elección función núcleo (SVM) y función de activación (GRBF)

En primer lugar, se ha de seleccionar la función núcleo (SVM) y la función de activación (GRBF) con la que se va a trabajar. En este caso, se ha optado por emplear en ambos casos funciones gaussianas, ya que en los dos métodos han presentado excelentes resultados en un gran número de aplicaciones y realizan la proyección a un espacio de características de infinitas dimensiones en el caso de la SVM. Está caracterizada por el parámetro sigma (σ), para la SVM, y por un vector de desviaciones típicas para la GRBF.

2.7.3. Entrenamiento y evaluación de la máquina

Se pueden distinguir dos etapas fundamentales en el diseño y evaluación de un clasificador máquina, una de entrenamiento y otra de evaluación. Se dispone de tres conjuntos de imágenes, que se corresponden con datos de entrenamiento, validación y test. En la primera fase, de entrenamiento, el objetivo es obtener los parámetros del clasificador. En esta fase se utilizan los dos primeros conjuntos de datos. Por tanto, se va a entrenar la máquina con el conjunto de entrenamiento para obtener los parámetros de la SVM o la GRBF y mediante los datos de validación, se obtienen mediante validación cruzada los parámetros óptimos en cada caso.

La validación cruzada es un proceso que sirve para tratar de evitar que se produzcan los problemas de sobreajuste y subajuste comentados en la Sección 2.7.1. Por tanto, una

vez elegida la función núcleo se deberá escoger debidamente para la SVM, el valor de los hiperparámetros C (ver Sección 2.6.2) y sigma (σ) (parámetro de núcleo). Y en el caso de la GRBF, el valor de sigma y la posición adecuada de los centroides (inicializaciones). En ambos casos, los parámetros de la máquina son calculados mediante validación cruzada, es decir, se escoge de entre un conjunto de valores los que proporcionan mejor resultado para los datos de validación. Dichos valores son los parámetros óptimos.

Por último, los datos de test se utilizan para evaluar finalmente la máquina con los parámetros óptimos obtenidos en la fase de entrenamiento para obtener la probabilidad de error final del sistema sobre datos que no se usaron de ninguna forma para obtener los parámetros de la máquina.

2.7.4. Coste computacional

El coste computacional del proceso de entrenamiento es elevado debido a la obtención de los parámetros óptimos. Aumenta progresivamente según el número de valores empleados para la obtención de los mismos.

2.8. Clasificación multiclase

Hasta el momento se ha considerado un problema de clasificación binario. Sin embargo, en situaciones reales se producen problemas de clasificación multiclase para considerar todas las clases a la vez [Weston y Watkins, 1998]. Debido a su elevado coste computacional surgen dos aproximaciones basadas en combinaciones de clasificadores binarios para tratar el caso multiclase. Se distinguen por tanto, dos métodos o estrategias de clasificación, uno contra todos y todos contra todos [Solera Ureña et al., 2006].

2.8.1. Uno contra todos

Se basa en comparar cada clase con todas las demás. Este método construye por tanto, k clasificadores, donde k es el número de clases. En cuanto al entrenamiento, cada modelo k -ésimo se entrenará con todas las clases, asignando una etiqueta positiva a los elementos de la clase k -ésima en cuestión y una etiqueta negativa al resto de elementos de las otras clases.

2.8.2. Todos contra todos

En este método se construyen $\frac{k(k-1)}{2}$ clasificadores. Consiste en comparar cada par de clases de forma separada, de tal forma que habrá un clasificador para cada posible par de clases. Por ello, es conocido también como el método uno contra uno. En este caso, en el entrenamiento se emplean únicamente los datos de las dos clases implicadas. En un problema de 8 clases tendrían lugar 28 clasificadores (1vs2, 1vs3, ..., 7vs8). Tras este proceso, se someten los datos de test a cada uno de esos clasificadores y se lleva a

cabo un proceso de votación que consiste en determinar qué clase ha obtenido más votos. Finalmente, la clase ganadora es la que obtiene el mayor número de votos.

En este trabajo se emplea la clasificación multiclase con la implementación del método todos contra todos tanto para la SVM como para la GRBF. A pesar de que el número de clasificadores a utilizar es mayor, al trabajar con menos datos se reduce el coste computacional proporcionando prestaciones similares al método uno contra todos [Hsu y Lin, 2002], y aumenta además la posibilidad de encontrar con mayor facilidad una frontera que separe ambas clases.

Capítulo 3

Requisitos y diseño de la solución

En este capítulo se presentan los requisitos del sistema y el esquema básico del clasificador describiendo para cada módulo los distintos métodos empleados para tratar de alcanzar la mejor solución posible.

3.1. Requisitos y especificaciones para la solución

Se quiere un clasificador que sea capaz de distinguir y clasificar lo mejor posible imágenes que pertenecen a diferentes clases. Teniendo en cuenta la función que debe desempeñar dicho clasificador se definen una serie de requisitos y especificaciones que debe cumplir el sistema a implementar:

- El aprendizaje tendrá lugar a partir de ejemplos mediante *métodos máquina* como máquinas de vectores soporte o redes neuronales.
- El clasificador deberá ser un *clasificador multiclase*, de tal forma que se puedan distinguir entre imágenes de diferentes clases (más de dos).
- Las tasas de clasificación deberán ser lo suficientemente elevadas como para proporcionar unas prestaciones adecuadas.
- El coste computacional de la implementación deberá ser acorde con las restricciones de tiempo (aplicación en tiempo real) empleando las tecnologías actuales.
- El sistema debe de ser capaz inicialmente de trabajar partiendo de imágenes de distinta resolución que provendrán del módulo de segmentación y deberán soportar la recepción de imágenes tanto en color como en blanco y negro.
- Deberá ser un sistema robusto frente al ruido y frente a distorsiones típicas como desplazamientos, desenfoque o cambios de iluminación. Es decir, las prestaciones deben mantenerse a pesar de que puedan variar ciertas condiciones.

3.2. Esquema básico

Teniendo en cuenta los requisitos presentados anteriormente, la Figura 3.1 muestra el diagrama de bloques que representa los distintos módulos funcionales del clasificador que se va a diseñar y evaluar.

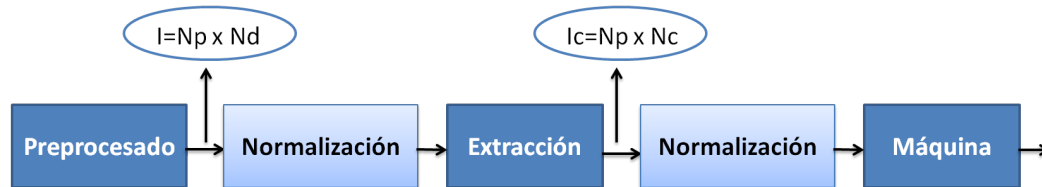


Figura 3.1: Representación del esquema básico del clasificador.

Como se puede observar en la figura existen cuatro módulos bien diferenciados: el módulo de preprocesado, extracción de características, normalización y el módulo máquina. A continuación se presentarán las principales características de cada módulo.

3.3. Preprocesado

Se corresponde con el primer bloque del esquema de clasificación. En él se llevan a cabo dos operaciones principalmente. Una de ellas consiste en transformar las imágenes en color a escala de grises para tratar de reducir así la dimensión del problema. La otra operación está basada en la resolución de las imágenes, ya que después del módulo de segmentación previo, llegan imágenes de cualquier resolución, por lo que es necesario convertir estas imágenes a una resolución fija a través de la cual trabajará el resto del sistema. Para ello, se estudiará a partir de una serie de resoluciones qué tamaño proporciona la menor probabilidad de error de los datos de entrenamiento. Sin embargo, dependiendo del problema, a veces es necesario también llevar a cabo otras operaciones, como por ejemplo aplicar un filtro de mediana para eliminar las deficiencias que presentan las imágenes u otras posibilidades que ofrece la literatura como la detección de bordes entre otras.

A la salida de este módulo, se obtiene una matriz de tamaño número de patrones por número de dimensión de la imagen.

3.4. Extracción de características

El objetivo de este módulo es obtener características que describan de la mejor manera posible el contenido de las imágenes. Dicha información deberá ser suficiente para distinguir unas imágenes de otras. Se evalúan distintos métodos de extracción de características, entre ellos, técnicas no supervisadas como PCA e ICA, en concreto el modelo ICA-ML y el algoritmo FastICA, y técnicas supervisadas como la de mínimos cuadrados y LDA.

El resultado es una matriz de tamaño número de patrones por número de características.

3.5. Normalización

Es un proceso que va a tener lugar en diferentes puntos, en concreto, tras los módulos de preprocesado y extracción de características. Mejora el funcionamiento tanto de la extracción de características (cuando la propia técnica no implica una normalización específica) como del clasificador máquina, ya que, al normalizar los datos, se transforman en otros valores que pertenecen a una escala predeterminada.

En el presente trabajo se probarán tres normalizaciones distintas y se verá cómo varían los resultados obtenidos. En particular, los datos serán normalizados por un lado en un rango de 0 a 1, de -1 a 1 y por varianza.

3.6. Clasificador máquina

Una vez que la máquina ha sido entrenada, el clasificador debe ser capaz de clasificar de forma correcta imágenes nuevas que no habían participado en dicho entrenamiento. Se van a emplear clasificadores de dos tipos. Se estudiarán las prestaciones del sistema empleando por un lado máquinas de vectores soporte, y por otro redes neuronales de tipo GRBF. En ambos casos, las funciones de activación empleadas son gaussianas.

Capítulo 4

Resultados y evaluación

En este capítulo se presentan los resultados obtenidos con los distintos métodos a partir de simulaciones. Se trata de analizar el funcionamiento en esta aplicación de los distintos métodos que se pueden implementar en cada módulo, con los respectivos parámetros generales que aparecen en el mismo. El número de posibles configuraciones es muy elevado (hay que tener en cuenta que hay varios métodos en cada módulo funcional, y que para cada método hay variables como número de píxeles de la etapa de preprocesado, número de características, número de neuronas en la red neuronal, etc.). Por tanto, como ya se dijo en el Capítulo 1, dadas las limitaciones temporales y de recursos que se tienen en un Trabajo Fin de Grado, no será posible simular todas las posibles configuraciones. En lugar de esto se realizará la simulación de un subconjunto de configuraciones que se ha seleccionado para poder extraer información suficientemente fiable sobre las principales tendencias de prestaciones que aparecen en cada módulo y en cada parámetro básico del mismo.

4.1. Generación de la base de datos para el estudio

Los métodos de aprendizaje máquina requieren una serie de patrones de ejemplo de los que aprender. En este trabajo, surge la necesidad de emplear imágenes que contengan señales de tráfico para el desarrollo de este Trabajo Fin de Grado. Éstas deben ser lo suficientemente representativas para que los resultados obtenidos sean relevantes. Tras una búsqueda de posibles bases de datos, se encontró una que se ajusta a lo que se está buscando para este trabajo [Escalera, 2009]. Se trata de una base de datos con 17 tipos de imágenes, algunas de las cuales son señales de tráfico. De esas 17 se opta por emplear únicamente 8 clases con señales de tráfico (Figura 4.1, Tabla 4.1).

Sin embargo, es una base de datos bastante reducida con una media de unas 24 imágenes por clase, por lo que era necesario extenderla aplicando distintas transformaciones sobre las imágenes. Es muy importante considerar qué transformaciones se van a aplicar a las imágenes, ya que esta decisión va a condicionar los resultados obtenidos a lo largo de todo el trabajo. Una vez elegidas las modificaciones y generada por tanto la base de datos, se va a trabajar con ella durante todo el estudio, de ahí su importancia.



Figura 4.1: Imágenes que representan cada clase empleada.

Clase	Señal
1	Calle sin salida
2	Velocidad máxima 20 km/h
3	Peligro peatones
4	Prohibido girar a la izquierda
5	Prohibido el paso
6	Ceda el paso
7	Prohibido parar y estacionar
8	Prohibido parar

Tabla 4.1: Descripción de las señales de cada clase.

En el presente trabajo se han aplicado tres transformaciones distintas a las imágenes. Por un lado se han realizado desplazamientos, que consisten en desplazar la imagen hacia arriba y hacia abajo, y hacia un lado u otro según el valor que se especifique. Es destacable que en ocasiones este tipo de desplazamientos se pueden producir en el módulo de segmentación. En segundo lugar, se han efectuado cambios de intensidad luminosa que varían las condiciones de iluminación de las imágenes apareciendo más claras o más oscuras según el valor de intensidad aplicado. Finalmente, también se ha aplicado *blurring* que se basa en desenfocar las imágenes produciendo la sensación de distorsión producida en muchos casos por movimientos de la cámara a la hora de captar las imágenes (ver Figura 4.2).

Las transformaciones tienen lugar progresivamente, es decir, se lleva a cabo la primera transformación sobre las imágenes ($24 \text{ imágenes} \times 25 \text{ transformaciones} = 600 \text{ imágenes transformadas}$) y la siguiente se realizará a partir de las imágenes obtenidas en la primera transformación ($(600 \text{ imágenes} \times 4 \text{ transformaciones}) + 600 \text{ imágenes transformadas} = 3000 \text{ imágenes transformadas}$). Lo mismo sucedería con la tercera transformación ($(3000 \text{ imágenes} \times 6 \text{ transformaciones}) + 3000 \text{ imágenes transformadas} = 21000 \text{ imágenes transformadas}$). De esta forma, se pasaría de tener únicamente 24 imágenes por clase, a tener 21.000 imágenes por cada clase.

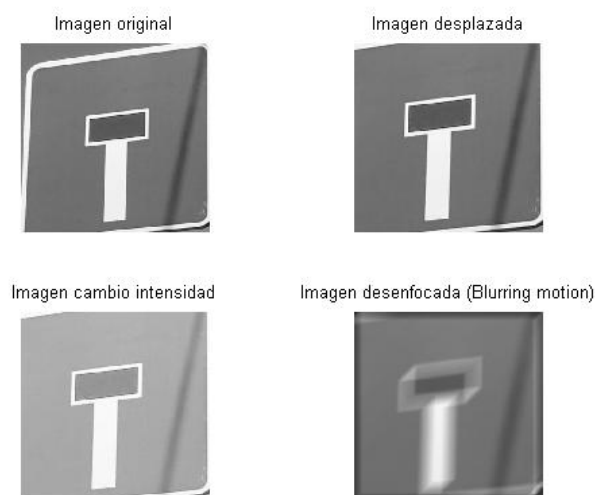


Figura 4.2: Imágenes con distintas transformaciones.

Hay que destacar que todas las transformaciones llevadas a cabo son aplicadas sobre las imágenes de alta resolución. Por otro lado, es importante considerar que las imágenes de la base de datos original no presentan todas la misma resolución.

4.2. Librerías empleadas

El entrenamiento de una SVM se resuelve inicialmente mediante un problema de programación cuadrática QP, computacionalmente costoso para bases de datos grandes. Existen implementaciones menos costosas computacionalmente. En este trabajo, se va a em-

plear la librería LIBSVM [Chang y Lin, 2012] para la clasificación, ya que funciona mejor con bases de datos grandes, es bastante más rápida.

Una de las técnicas de extracción de características empleadas es LDA cuya toolbox se puede encontrar publicada en [van der Maaten, 2012], y su autor es Laurens van der Maaten. Por otro lado, las rutinas para implementar las técnicas ICA ML y Fast ICA se pueden obtener de [Hansen et al., 2006] y de [Hyvärinen et al., 2005], respectivamente.

Por otro lado, las rutinas de Matlab empleadas para el procesamiento de imagen se pueden encontrar en la toolbox [MathWorks, 2012].

4.3. Resultados sin ruido

En esta sección, se presentan los resultados obtenidos para los distintos métodos empleando las imágenes de la base de datos sin ruido. Hay que considerar para ello el tamaño de los distintos conjuntos con los que se va a trabajar, siendo de 420 imágenes para el conjunto de entrenamiento (1 imagen de cada 50), de 420 imágenes (1 de cada 50, distintas de las de entrenamiento) para el de validación y las imágenes restantes, 20.160 para el conjunto de test.

4.3.1. Resolución de imágenes

En primer lugar, se desea obtener, a partir de la imagen proporcionada por el módulo previo de segmentación (modelado por las imágenes de la base de datos, de distinta resolución), una imagen de tamaño fijo, y deseablemente con la menor resolución posible que proporcione unas prestaciones razonables. Ese cambio de resolución forma parte del módulo de preprocesado y se fijará para el resto de módulos a partir de ahora. Es importante considerar el coste que conlleva cambiar la resolución a toda la base de datos, ya que implica evaluar distintas resoluciones en los 28 clasificadores. Partiendo de que se dispone de 28 clasificadores se escogen dos casos, uno bueno (pareja 1) y otro malo (pareja 2). En concreto la pareja 1 (ver Figura 4.3) se corresponde con el caso bueno (relativamente sencillo), es decir, las imágenes entre ambas clases poseen grandes diferencias y no deberían presentar problemas a la hora de clasificarse (señales cuadradas frente a triangulares), y la pareja 2 (ver Figura 4.4) que se corresponde con el caso malo (caso más complicado), es decir, las imágenes presentan un cierto grado de parecido (ambas señales de forma circular).

Se valora cuál sería la probabilidad de error para los casos bueno y malo empleando como método de extracción de características PCA y normalización de los datos por varianza, y en base a ello se elige la resolución óptima. Dicha resolución es obtenida mediante validación cruzada entre un conjunto de valores (4×4 , 8×8 , 12×12 , 16×16 , 24×24 , 32×32) a través del conjunto de validación. Se realiza por tanto el entrenamiento de la máquina para cada uno de esos valores y la resolución óptima vendrá determinada por la mínima probabilidad de error del conjunto de validación.

Los resultados que se muestran en esta sección han sido realizados con máquinas de vectores soporte y cuyo resultado obtenido se fijará para el resto de estudios a lo largo

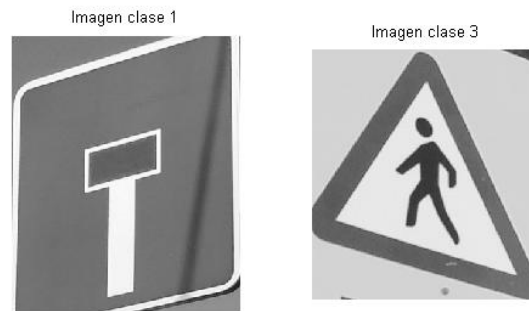


Figura 4.3: Imágenes representativas de las clases 1 y 3 (Pareja 1 - Caso “bueno”).

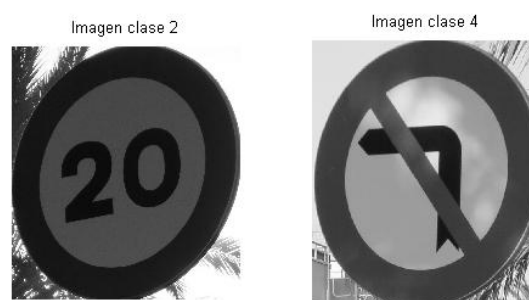


Figura 4.4: Imágenes representativas de las clases 2 y 4 (Pareja 2 - Caso “malo”).

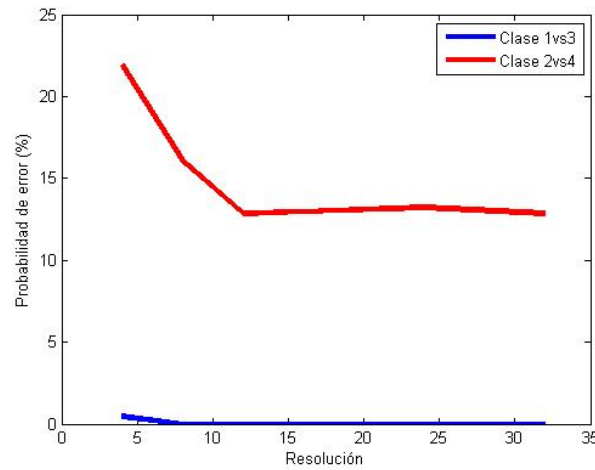


Figura 4.5: Representación de la probabilidad de error en función de la resolución para obtener el número óptimo de resolución para la SVM.

de las siguientes pruebas. En la Tabla 4.2 se recogen las probabilidades de error en % obtenidas para cada pareja de clases, aplicando diferentes cambios de resolución a las imágenes de cada clase.

	Res. 4×4	Res. 8×8	Res. 12×12	Res. 16×16	Res. 24×24	Res. 32×32
Pareja 1	0.47	0.00	0.00	0.00	0.00	0.00
Pareja 2	21.90	16.07	12.85	12.97	13.21	12.85

Tabla 4.2: Probabilidades de error obtenidas para distintas resoluciones para SVM.

Para obtener el valor óptimo de resolución se tiene en cuenta el peor caso. Por tanto, si se analizan los resultados y se considera la Figura 4.5, una resolución de 4×4 sería insuficiente, ya que la probabilidad de error que presenta la pareja 2 es muy elevada, como se puede observar en la Tabla 4.2. Por otro lado, la menor probabilidad de error que se obtiene es de 12.857% cuyo valor coincide para las resoluciones de 12×12 y de 32×32 . A la vista de los resultados resulta obvio que el valor óptimo será de 12×12 , ya que presenta los mismos resultados que 32×32 pero con una resolución menor. Dicho valor es calculado mediante validación cruzada obteniéndose para la SVM un valor óptimo de 12×12 píxeles.

En este caso, como se comentaba al principio de esta sección se ha empleado el método de extracción de características PCA y una normalización por varianza. Sin embargo, los resultados que se obtienen con otras configuraciones son similares, no hay grandes diferencias entre métodos, por lo que se decide seguir en adelante con esta configuración debido a las restricciones de tiempo y recursos disponibles.

4.3.2. Métodos de normalización

En esta sección se evalúan los resultados obtenidos para SVM con una resolución de 12×12 , tras emplear 3 tipos de normalización distintos, con un número de características prefijado (en este caso 3 características) y empleando como método de extracción de características PCA. El objetivo es comparar las prestaciones que presenta cada uno de los métodos de normalización.

En primer lugar, se realiza una normalización de los datos de 0 a 1 obteniendo una probabilidad de acierto promedio de 82.8336 %. Cuando la normalización tiene lugar entre -1 y $+1$, la probabilidad de acierto es algo menor, 80.4638 %. El último tipo de normalización probado es la normalización por varianza cuya probabilidad de acierto es todavía menor que las dos anteriores, de 78.6663 %.

Tras ver los resultados, se puede comprobar que la normalización de 0 a 1 funciona mejor que el resto de métodos probados. De ahora en adelante, para el resto de pruebas se empleará la normalización de 0 a 1. La Tabla 4.3 muestra los resultados obtenidos empleando dicha normalización mediante una tabla de contingencia. Las filas de la tabla indican que siendo las imágenes de la clase a, el sistema se equivoque y diga que pertenecen a la clase b. Es decir, indican la probabilidad de error que comete el sistema a la hora de clasificar las imágenes. Por otro lado, la diagonal principal de la tabla indica cuál es la probabilidad de acierto obtenida por cada clase.

	Clase 1	Clase 2	Clase 3	Clase 4	Clase 5	Clase 6	Clase 7	Clase 8
Clase 1	96.731	0.179	0.000	0.000	0.104	0.000	0.561	0.045
Clase 2	0.114	57.684	0.069	4.182	0.228	0.714	5.253	11.478
Clase 3	0.005	0.521	94.812	0.094	0.124	0.283	0.134	1.200
Clase 4	0.000	4.058	0.005	74.182	0.759	0.050	6.468	0.774
Clase 5	0.074	0.769	0.000	0.139	85.809	5.347	1.017	1.017
Clase 6	0.689	0.769	0.000	0.084	0.734	88.095	1.563	1.572
Clase 7	0.337	3.418	0.050	0.337	0.089	0.060	84.033	3.041
Clase 8	0.000	6.915	0.109	0.025	0.605	1.463	2.852	81.324

Tabla 4.3: Tabla de contingencia empleando normalización de 0 a 1 para la SVM.

4.3.3. Número de características

En esta sección se pretende obtener el número óptimo de características para la SVM, para ello se calcula dicho valor por cada pareja de clases mediante validación cruzada, empleando como método de extracción PCA, con una normalización de los datos de 0 a 1 y una resolución de imagen de 12×12 píxeles. Se han realizado pruebas para 15 valores distintos de características en concreto, de 1 a 10, 15, 25, 50, 80 y 100 características. Por tanto, para cada uno de esos valores se realiza el entrenamiento de la máquina para obtener a través del conjunto de entrenamiento los parámetros de la SVM. Posteriormente

mediante los datos de validación se obtienen los parámetros óptimos de la máquina (el número de características óptimo, entre ellos), y por último con los datos de test y considerando dichos parámetros se procede a evaluar la máquina para obtener la probabilidad de error final. En ningún caso para esta configuración han sido necesarios los valores de 50, 80 y 100 características (ver información correspondiente a PCA en las Tablas 4.4 y 4.5).

Es importante elegir bien el número de características adecuado para cada clase, ya que es un factor determinante para el resultado de nuestro sistema y también importante a la hora de reducir recursos. De esta forma, cada clase se ciñe al número mínimo de características necesario.

4.3.4. Comparación de técnicas de extracción para SVM y GRBF

El principal objetivo es realizar una comparativa de los distintos métodos de extracción empleados para máquinas de vectores soporte y redes neuronales. Se tienen en consideración por un lado las técnicas de extracción no supervisadas frente a las técnicas supervisadas, para evaluar su comportamiento.

Una vez fijada la resolución de las imágenes, el tipo de normalización que se va a emplear y el número de características óptimo, se analizan las prestaciones de las distintas técnicas de extracción.

Número de características óptimo

Las Figuras 4.6, 4.7, 4.8 y 4.9 muestran la representación del número de características óptimo para cada método de extracción (PCA, ICA ML, Fast ICA y mínimos cuadrados) para SVM considerando dos casos, uno relativamente sencillo y otro más complicado, de nuevo para las parejas 1 y 2 citadas en la Sección 4.3.1.

Hay que destacar, que los métodos PCA, Fast ICA y mínimos cuadrados han sido probados para los números de características de 1 a 10, 15, 25, 50, 80 y 100. Sin embargo, para ICA ML por problemas de memoria los valores empleados han sido de 1 a 10, 15, 20, 25 y 30 características. Por otro lado, en la técnica LDA se ha fijado el número de características a una, ya que el número de características a emplear en este método está limitado al número de clases (2 clases en un clasificador binario) - 1.

Para tener una idea más general y considerando por ello, todos los casos posibles, la Tabla 4.4 muestra un resumen acerca del número medio, el número mínimo y el número máximo de características teniendo en cuenta todas las parejas de clases para SVM.

Como puede observarse en la Tabla 4.4, el método de extracción de mínimos cuadrados obtiene un número medio de características menor que el resto de técnicas, y como valor mínimo obtiene un valor de una única característica. Sin embargo, por algún motivo, alguna pareja de clases requiere un valor elevado de características, 80 en concreto (valor máximo de la tabla). Por este motivo, se presenta la tabla resumen 4.5 con la misma información que proporciona la Tabla 4.4, pero esta vez mostrando los resultados para cada clase en concreto. De esta forma, se detectará qué clases están demandando un valor de características tan elevado.

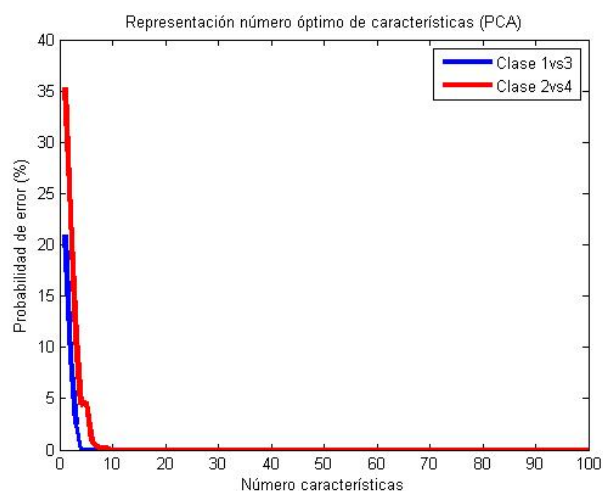


Figura 4.6: Representación de la probabilidad de error en función del número de características empleando como método de extracción PCA para SVM.

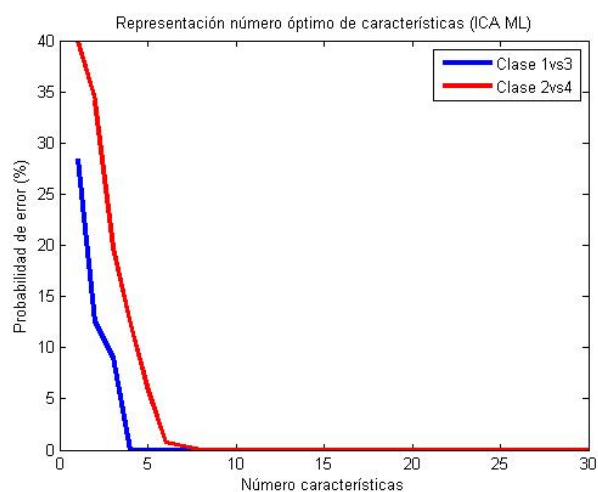


Figura 4.7: Representación de la probabilidad de error en función del número de características empleando como método de extracción ICA ML para SVM.

	PCA	ICA ML	Fast ICA	Mín. Cuadrados	LDA
Número medio	6.64	5.67	6.46	4.29	1
Mínimo	3	3	3	1	1
Máximo	25	9	15	80	1

Tabla 4.4: Tabla resumen con el número de características general requerido por cada método para la SVM.

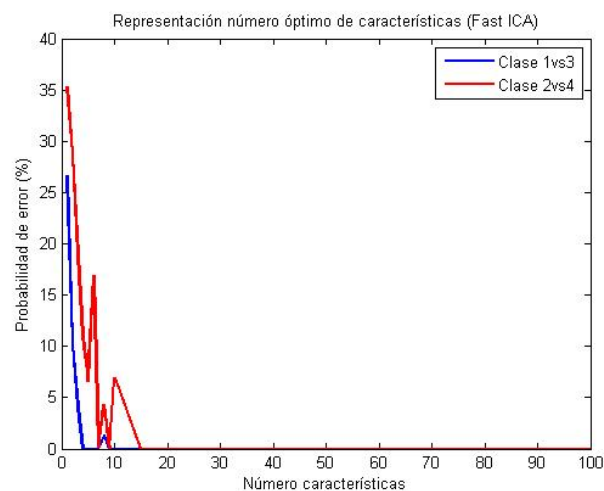


Figura 4.8: Representación de la probabilidad de error en función del número de características empleando como método de extracción Fast ICA para SVM.

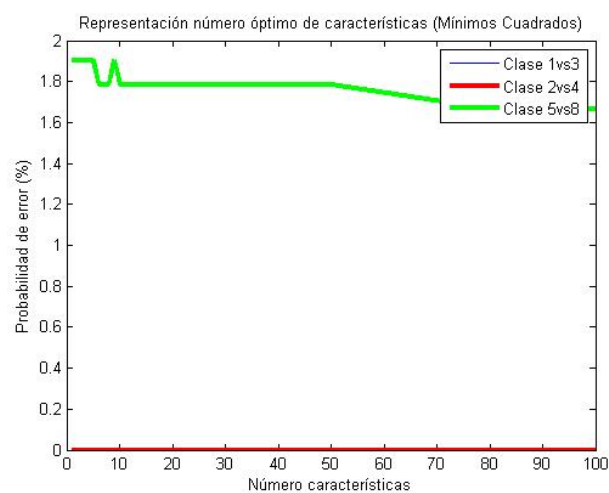


Figura 4.9: Representación de la probabilidad de error en función del número de características empleando como método de extracción Mínimos Cuadrados para SVM.

		C 1	C 2	C 3	C 4	C 5	C 6	C 7	C 8
PCA	N.medio	4	10.57	4	5.85	7	6.14	8.28	8.71
	Mín.	3	4	3	3	4	4	3	3
	Máx.	6	25	5	10	10	10	25	25
ICA ML	N.medio	4	6.57	4.57	5.57	6.42	6	6	5.85
	Mín.	3	4	3	3	4	4	4	3
	Máx.	6	9	6	8	8	8	7	7
Fast ICA	N.medio	3.71	8.85	5.71	6.28	6.28	7.57	5.85	6.28
	Mín.	3	4	3	3	4	5	4	3
	Máx.	5	15	9	9	8	15	7	8
Min Cuadr	N.medio	1	2.42	1.42	1	12.28	1	1.57	12.42
	Mín.	1	1	1	1	1	1	1	1
	Máx.	1	10	4	1	80	1	4	80
LDA	N.medio	1	1	1	1	1	1	1	1
	Mín.	1	1	1	1	1	1	1	1
	Máx.	1	1	1	1	1	1	1	1

Tabla 4.5: Tabla resumen con el número de características por clases requerido por cada método para la SVM.

Como se observa en la Tabla 4.5, para la técnica de mínimos cuadrados, las clases 5 y 8 requieren un mayor número de características que el resto. La Figura 4.9 muestra la curva de probabilidad de error frente al número de características para este caso concreto (clase 5 frente a 8). En dicha gráfica se puede comprobar que la probabilidad de error obtenida es siempre 0 tanto para el caso bueno como para el malo, salvo para la pareja conflictiva que obtiene probabilidades de error superiores a 0, entorno a 1,7 % y 1,9 %. Como se puede observar en la gráfica, la probabilidad de error obtenida empleando de 10 a 50 características es la misma, por lo que habría que plantearse si la mejora empleando 80 características es relevante. En este caso, la diferencia es muy pequeña por lo que no sería necesario emplear un número tan elevado.

Por otro lado, comparando los resultados gráficamente, si se consideran las gráficas 4.6, 4.7, 4.8 y 4.9 se puede verificar que la técnica Fast ICA tiene un comportamiento menos monótono que el resto de gráficas. Este comportamiento se debe a que la solución de esta técnica depende de inicializaciones, es decir, presenta el problema de los mínimos locales.

Número de neuronas óptimo

Hasta ahora los resultados proporcionados en apartados anteriores se basaban en la SVM donde no era necesario calcular el número de vectores soporte porque éste se obtenía de forma automática. Sin embargo, en GRBF no sucede así y por tanto, hay que calcular el número de neuronas requerido mediante validación cruzada.

Se emplea una resolución de imágenes fija (12×12), normalización de 0 a 1 y el número

de características óptimo que se obtuvo para SVM, y se procede a calcular el número de neuronas óptimo para cada pareja de clases. El número óptimo de neuronas se obtiene al igual que el número de características mediante validación cruzada entre un conjunto de valores posibles. En este caso, se han probado 11 valores (de 1 a 10 y 15 neuronas). Sin embargo, en la técnica LDA se ha fijado el número de neuronas a una, ya que tras probar con un número mayor el resultado apenas variaba, por lo que no compensaba incrementar la complejidad de la red.

Las Tablas 4.6 y 4.7 muestran información esta vez sobre las neuronas empleadas en cada uno de los métodos de extracción. Se muestra el número medio, máximo y mínimo de neuronas requeridas para cada técnica. En este caso, el número medio de neuronas para mínimos cuadrados es menor que para el resto de métodos según la Tabla 4.6. Dentro de los no supervisados ICA ML muestra los mejores resultados.

	PCA	ICA ML	Fast ICA	Mín. Cuadrados	LDA
Número medio	5.43	5.03	7.46	2.75	1
Mínimo	2	2	2	1	1
Máximo	10	9	15	10	1

Tabla 4.6: Tabla resumen con el número de neuronas general requerido por cada método para la GRBF.

		C 1	C 2	C 3	C 4	C 5	C 6	C 7	C 8
PCA	N.medio	3.28	6	5.14	6.42	5	5	6.28	6.14
	Mín.	2	5	2	4	2	2	2	3
	Máx.	5	8	8	9	8	8	10	10
ICA ML	N.medio	6.57	6.28	2.71	4.28	4.42	5.57	5.28	4.85
	Mín.	3	4	2	2	2	2	2	2
	Máx.	9	9	4	9	9	8	8	8
Fast ICA	N.medio	8.57	8.85	4.42	6.42	6	9.42	9.42	8
	Mín.	2	4	2	4	2	3	2	2
	Máx.	15	15	7	10	15	15	15	15
Min Cuadr	N.medio	2.14	2.71	2.14	1.57	3	2.14	3.28	4.71
	Mín.	1	1	1	1	1	1	2	2
	Máx.	3	7	4	4	10	4	5	10
LDA	N.medio	1	1	1	1	1	1	1	1
	Mín.	1	1	1	1	1	1	1	1
	Máx.	1	1	1	1	1	1	1	1

Tabla 4.7: Tabla resumen con el número de neuronas por clases requerido por cada método para la GRBF.

Las Figuras 4.10, 4.11, 4.12 y 4.13 muestran la obtención del número de neuronas óptimo para los dos casos mencionados en apartados anteriores, un caso relativamente

sencillo formado por la pareja 1 y otro más complicado representado por la pareja 2. Si se comparan las gráficas se puede comprobar que de nuevo la técnica Fast ICA presenta una curva menos monótona que el resto, debido a las distintas inicializaciones como se comentó al obtener el número de características óptimo para cada una de las técnicas.

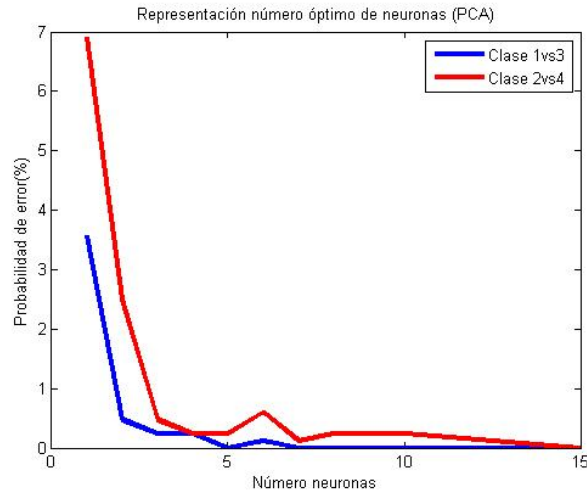


Figura 4.10: Representación de la probabilidad de error en función del número de neuronas empleando como método de extracción PCA para GRBF.

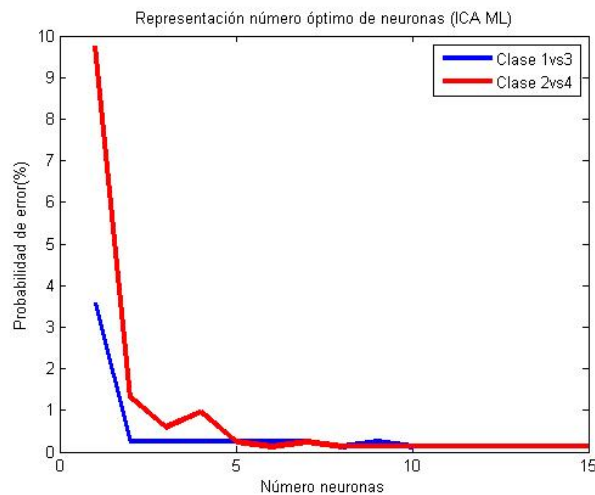


Figura 4.11: Representación de la probabilidad de error en función del número de neuronas empleando como método de extracción ICA ML para GRBF.

Comparación de técnicas de extracción de características

Los resultados presentados en esta sección provienen de emplear imágenes sin ruido con una resolución de 12×12 píxeles tanto en el entrenamiento como en la evaluación.

La Tabla 4.8 refleja los resultados obtenidos en estas circunstancias para SVM y GRBF con cada una de las técnicas de extracción de características. Como se puede observar

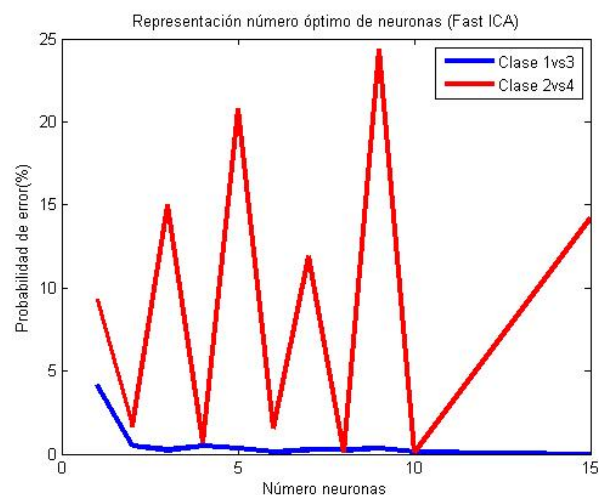


Figura 4.12: Representación de la probabilidad de error en función del número de neuronas empleando como método de extracción Fast ICA para GRBF.

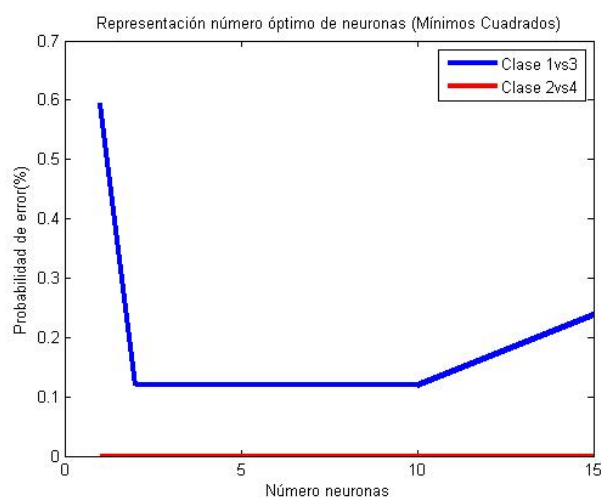


Figura 4.13: Representación de la probabilidad de error en función del número de neuronas empleando como método de extracción Mínimos Cuadrados para GRBF.

para SVM las probabilidades de acierto obtenidas son similares para todos los métodos no supervisados, sin embargo, el algoritmo Fast ICA presenta una probabilidad de acierto algo superior que el resto. Se puede comprobar también que la probabilidad obtenida para el método de mínimos cuadrados es menor que para el resto de técnicas no supervisadas, esta situación puede deberse al problema planteado en el Apartado 4.3.4.

	PCA	ICA ML	Fast ICA	Mín. Cuadrados	LDA
SVM	99.00	99.192	99.197	98.73	99.76
GRBF	98.00	98.12	98.78	99.03	99.42

Tabla 4.8: Probabilidades de acierto promedio obtenidas con cada técnica para SVM y GRBF empleando imágenes sin ruido.

Para poder realizar una comparación entre técnicas supervisadas para SVM, se emplea en ambas técnicas (mínimos cuadrados y LDA) una única característica. Los resultados obtenidos se muestran en la Tabla 4.9, donde se puede observar que las prestaciones a igualdad de condiciones son ligeramente superiores con LDA.

	Mín. Cuadrados	LDA
Prob. acierto promedio (%)	98.84	99.76

Tabla 4.9: Tabla comparativa de las técnicas supervisadas con una única característica para la SVM.

Para GRBF de nuevo la técnica no supervisada que mejor resultado proporciona es Fast ICA, mientras que como técnica supervisada, LDA destaca frente a las demás (ver Tabla 4.8). La Tabla 4.10 refleja cuál sería el resultado del método de mínimos cuadrados cuando se emplea también una neurona, en este caso los dos métodos presentan el mismo resultado.

	Mín. Cuadrados	LDA
Prob. acierto promedio (%)	99.42	99.42

Tabla 4.10: Tabla comparativa de las técnicas supervisadas con una única neurona para GRBF.

Resultados aplicando filtro de mediana

A pesar de que sin ruido no tiene mucho sentido o no es útil emplear un filtro de mediana, el objetivo principal que se persigue en esta prueba es la comparación con estudios posteriores. Se emplean por tanto imágenes sin introducir ruido con una resolución de 12×12 aplicando sin embargo, un filtro de mediana sobre las imágenes de alta resolución. La finalidad de esta prueba, se basa en tratar las imágenes “sin ruido”, eliminando mediante un filtro de mediana posibles componentes espurios que pueden estar presentes en las imágenes.

La Tabla 4.11 presenta los resultados obtenidos con cada una de las técnicas de extracción de características tanto para SVM como para GRBF. En este caso, los resultados obtenidos para SVM están por debajo de los valores especificados en la Sección 4.3.4, salvo la técnica LDA que proporciona un valor similar. Sin embargo, para GRBF los resultados son ligeramente superiores que en el caso de no utilizar el filtro de mediana.

	PCA	ICA ML	Fast ICA	Min.Cuadrados	LDA
SVM	98.99	99.17	99.19	98.81	99.80
GRBF	98.78	98.56	98.68	98.96	99.44

Tabla 4.11: Probabilidades de acierto promedio obtenidas con cada técnica para SVM y GRBF en imágenes sin ruido utilizando un filtro de mediana en el preprocesado.

4.3.5. SVM vs GRBF

Como se puede observar en el Apartado 4.3.4 los resultados obtenidos empleando técnicas no supervisadas son mejores para SVM que para la GRBF, mientras que para técnicas supervisadas sucede al contrario, las prestaciones de la GRBF son ligeramente superiores.

La Tabla 4.12 muestra una comparativa en cuanto a complejidad, teniendo en cuenta el número de parámetros necesarios para la SVM y la GRBF empleando imágenes sin ruido sin aplicar filtro de mediana. Conforme mayor sea el número de parámetros, la memoria requerida también será mayor, y también lo será el tiempo requerido para la evaluación.

En el caso de la SVM el número de parámetros viene determinado por el número de características y el número de vectores soporte, mientras que para la GRBF se determina a través del número de características y de neuronas.

Por regla general, el número de neuronas suele ser bastante menor que el de vectores soporte. Como se observa en la Tabla 4.12, el número de parámetros es bastante mayor para la SVM que para la GRBF.

	PCA	ICA ML	Fast ICA	Mín. Cuadrados	LDA
Núm. Parámetros SVM	58916	30311	50702	44087	10500
Núm. Parámetros GRBF	2244	1825	2857	1999	112

Tabla 4.12: Parámetros necesarios para SVM y GRBF con imágenes sin ruido sin aplicar filtro de mediana.

4.4. Resultados con ruido

Para poder realizar una comparativa lógica entre los resultados con ruido y sin ruido, las condiciones aplicadas en ambos casos tienen que ser las mismas. Por ello, inicialmente se fija la resolución a 12×12 píxeles (la misma que en la Sección 4.3.1), se aplica una normalización a los datos de 0 a 1 y se obtiene el número de características y de neuronas del mismo modo que en la Sección 4.3. De esta forma, la comparación tiene sentido, es decir, ambos casos están en igualdad de condiciones, lo único que varía son las imágenes que presentan ruido *sal y pimienta* con una densidad de ruido de 0.05.

A continuación, se van a presentar varios resultados dependiendo de si únicamente los datos de test presentan ruido o si se han empleado imágenes ruidosas en el entrenamiento.

4.4.1. Entrenamiento sin ruido

En esta sección se presentan los resultados obtenidos en cada una de las pruebas realizadas para la SVM y la GRBF empleando para el entrenamiento imágenes sin ruido.

Comparación de técnicas de extracción de características

A continuación, se muestran los resultados obtenidos empleando imágenes sin ruido para el entrenamiento y con ruido para la evaluación.

1. Entrenamiento sin ruido y evaluación con ruido.

Esta prueba consiste en realizar el entrenamiento empleando imágenes sin ruido e imágenes con ruido sin aplicar filtro de mediana para la evaluación. Por lo que los resultados deberían ser peores que en la Sección 4.3.

La Tabla 4.13 presenta los resultados obtenidos con cada una de las técnicas para SVM y GRBF. Se puede comprobar que los resultados obtenidos bajo las condiciones de este apartado son peores que los que se obtenían en la Sección 4.3.

	PCA	ICA ML	Fast ICA	Min.Cuadrados	LDA
SVM	98.80	99.02	98.94	98.41	99.63
GRBF	97.78	97.85	98.53	98.75	99.42

Tabla 4.13: Probabilidades de acierto promedio obtenidas con cada técnica para SVM y GRBF empleando una densidad de ruido de 0.05.

Hasta ahora, los resultados proporcionados en la Sección 4.4 son empleando una densidad de ruido de 0.05. Sin embargo, se va a llevar a cabo un estudio para ver qué pasaría si variase la densidad del ruido para SVM.

Como se puede observar en la Tabla 4.14, a medida que aumenta la densidad del ruido, disminuyen progresivamente las prestaciones. Sin embargo, si se tiene en cuenta la Figura 4.14 la degradación visual de las imágenes es mayor que la degradación que

	PCA	ICA ML	Fast ICA	Mín. Cuadrados	LDA
Densidad 0	99.00	99.12	99.19	98.73	99.76
Densidad 0.05	98.80	99.02	98.94	98.41	99.63
Densidad 0.1	97.66	98.73	98.47	98.12	99.46
Densidad 0.25	89.16	96.08	93.70	96.34	98.60

Tabla 4.14: Probabilidades de acierto para distintos valores de densidad de ruido para la SVM.

sufren las prestaciones. Hay que destacar que no todos los métodos de extracción sufren la misma degradación, sino que las técnicas ICA ML entre las técnicas no supervisadas y LDA entre las técnicas supervisadas son más robustas.

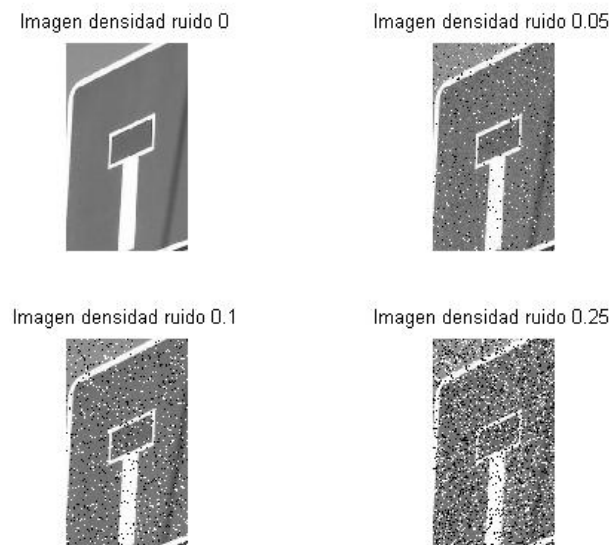


Figura 4.14: Degradación de imágenes según la densidad del ruido.

Para la GRBF, los resultados son peores que los especificados en la Sección 4.3.4, salvo utilizando la técnica LDA que proporciona prácticamente el mismo resultado.

4.4.2. Entrenamiento con ruido

En esta sección se presentan los resultados obtenidos en cada una de las pruebas realizadas para la SVM y la GRBF empleando también para el entrenamiento imágenes con ruido.

Comparación de técnicas extracción de características

A continuación, se muestran los resultados para los dos estudios realizados. En una de las pruebas se emplean imágenes con ruido aplicando filtro de mediana y en la otra,

imágenes con ruido sin aplicar filtro de mediana.

1. Imágenes con ruido aplicando filtro de mediana

En este caso, se aplica sobre las imágenes de la base de datos un ruido de sal y pimienta con una densidad de 0.05. Para subsanar dicho ruido se emplea un filtro de mediana con el objetivo de proporcionar resultados coherentes a los que se obtenían anteriormente en la Sección 4.3. Por un lado, se podría esperar que los resultados en esta situación sean peores debido al ruido introducido. Pero por otro lado cabe también pensar que entrenar la máquina con imágenes más parecidas a aquellas que se van a presentar en la evaluación (ruidosas) puede ser beneficioso.

La Tabla 4.15 muestra los resultados obtenidos con cada técnica de extracción de características tanto para SVM como para GRBF. Comparando estos resultados con los de la Sección 4.3, se obtiene que para imágenes con ruido el resultado es mejor, salvo con PCA cuyo resultado permanece casi invariable. Puede suceder que las imágenes de la base de datos contengan componentes espurios y al emplear un filtro de mediana para eliminar el ruido mejore notablemente el aspecto de las imágenes, proporcionando resultados superiores a los que se obtendrían empleando imágenes sin ruido.

La Figura 4.15 muestra cómo afecta el aplicar un filtro de mediana a imágenes que presentan blurring. A simple vista, el ruido de sal y pimienta se elimina sin quedar rastro, sin embargo hay que estudiar si también se mejoran las prestaciones de la imagen internamente y no de forma visual únicamente.

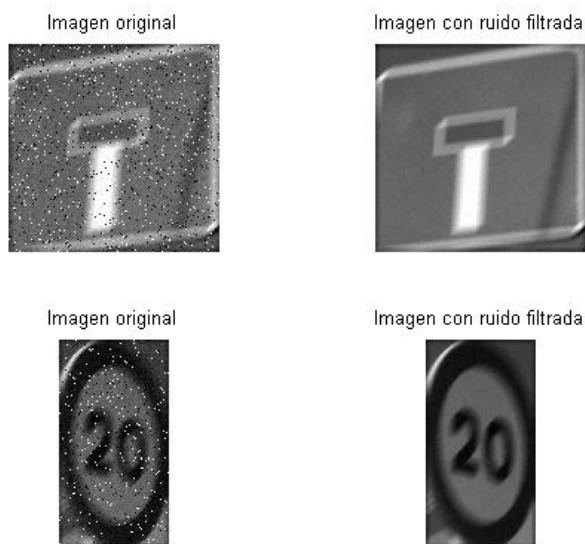


Figura 4.15: Representación de cómo afecta el filtro de mediana a imágenes con blurring.

Para GRBF, al igual que sucedía con la SVM, los resultados obtenidos empleando técnicas no supervisadas son ligeramente superiores a los resultados sin ruido obtenidos en la Sección 4.3. Sin embargo, la técnica supervisada de mínimos cuadrados proporciona una probabilidad de acierto algo menor que antes.

	PCA	ICA ML	Fast ICA	Min.Cuadrados	LDA
SVM	99.41	99.49	99.49	98.76	99.73
GRBF	99.20	99.20	99.12	98.81	99.54

Tabla 4.15: Probabilidades de acierto promedio obtenidas con cada técnica para SVM y GRBF en imágenes ruidosas utilizando un filtro de mediana.

2. Imágenes con ruido sin aplicar filtro de mediana

El experimento consiste en presentar las imágenes con el ruido sin eliminarlo previamente con un filtro de mediana. De esta forma, se comprueba cuál es el efecto que producen las imágenes con el ruido sin emplear ningún tipo de técnica para tratarlo. La Tabla 4.16 representa los resultados obtenidos con cada técnica de extracción para SVM y GRBF. Los resultados no distan mucho de los obtenidos anteriormente, siguen siendo superiores a los resultados sin ruido.

Para la GRBF sucede lo mismo que en la prueba para la SVM, es decir, los resultados son algo mejores en este caso que en el caso de emplear imágenes sin ruido. De nuevo para mínimos cuadrados sigue siendo peor.

	PCA	ICA ML	Fast ICA	Min.Cuadrados	LDA
SVM	99.34	99.31	99.42	98.60	99.68
GRBF	98.68	99.06	99.19	98.67	99.45

Tabla 4.16: Probabilidades de acierto promedio obtenidas con cada técnica para SVM y GRBF en imágenes ruidosas sin filtro de mediana.

4.4.3. Resolución de imágenes

Hasta el momento para los resultados presentados con ruido se ha empleado una resolución de imagen de 12×12 píxeles, la misma que la que se empleó para imágenes sin ruido. Sin embargo, esa resolución no tiene por qué ser la óptima en este caso también. Por ese motivo, se opta a calcular cómo se modificaría la resolución óptima en presencia de ruido. Los resultados facilitados en esta sección a partir de ahora, provienen de aplicar un ruido de *sal y pimienta* de densidad 0.05 sobre las imágenes de la base de datos. El procedimiento a emplear es el mismo que para las imágenes sin ruido, con la única diferencia de tener que solventarlo empleando filtros de mediana tratando de proporcionar resultados similares a los que se obtenían anteriormente en la Sección 4.3.

En este caso, el tamaño de resolución requerido para las imágenes es algo mayor que el que obtuvimos en la Sección 4.3.1, donde no había ruido en las imágenes. De nuevo la Tabla 4.17 refleja las probabilidades de error obtenidas para un caso relativamente sencillo y otro más complicado, empleando para ello las mismas parejas de clases y las mismas restricciones que las especificadas en la Sección 4.3.1. Tras analizar los resultados de la tabla, resulta conveniente emplear una resolución de 16×16 . Para ello es necesario tener en cuenta el compromiso entre la probabilidad de error de ambas parejas y el tamaño

de las imágenes. Aunque la probabilidad de error para el caso malo se consigue con una resolución de 8×8 , se trata de obtener para el caso bueno una probabilidad de 0 % y esto sucede para una resolución de 16×16 (Figura 4.16).

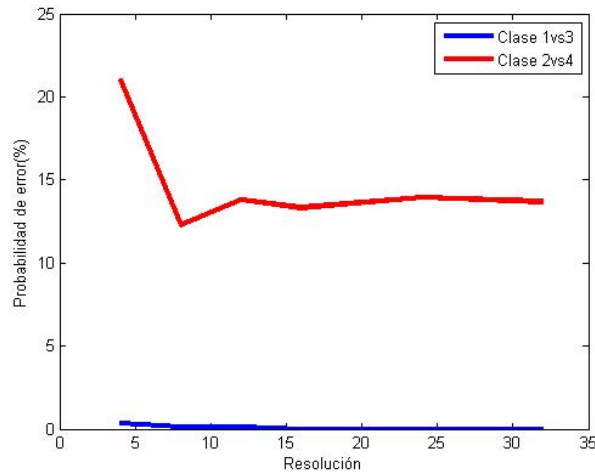


Figura 4.16: Representación de la probabilidad de error en función de la resolución para obtener el número óptimo de resolución para la SVM.

	Res. 4×4	Res. 8×8	Res. 12×12	Res. 16×16	Res. 24×24	Res. 32×32
Pareja 1	0.35	0.12	0.12	0.00	0.00	0.00
Pareja 2	21.07	12.26	13.80	13.33	13.92	13.69

Tabla 4.17: Probabilidades de error obtenidas para distintas resoluciones con imágenes ruidosas para la SVM.

4.4.4. Métodos de normalización

Se llevan a cabo los mismos tipos de normalización que los nombrados en la Sección 4.3.2. En este caso, la probabilidad acierto promedio obtenida empleando una normalización de 0 a 1 es de 82.3562 %. Si la normalización se realiza de -1 a 1, el resultado es de 78.3724 % y por último, cuando se emplea la normalización por varianza la probabilidad de acierto es de 80.7403 %. De nuevo, al igual que sucedía con los datos sin introducir ruido se obtiene mejor resultado empleando la normalización de 0 a 1. La Tabla 4.18 muestra la tabla de contingencia obtenida con este tipo de normalización.

4.4.5. Comparación de técnicas de extracción para SVM y GRBF

En esta sección se presentan los resultados para las distintas técnicas de extracción empleadas y se realiza la comparación entre ellas, considerando lo expuesto en la Sección 4.3.4.

	Clase 1	Clase 2	Clase 3	Clase 4	Clase 5	Clase 6	Clase 7	Clase 8
Clase 1	96.4038	0.0099	0.000	0.000	0.0843	0.0149	1.4583	0.0496
Clase 2	0.2877	59.3849	0.00198	6.2351	0.6994	0.6994	3.7897	10.1389
Clase 3	0.0397	0.1935	94.6478	0.0694	0.0546	0.2579	0.4018	1.1954
Clase 4	0.000	8.4077	0.000	68.8145	0.9970	0.0794	6.5476	0.3224
Clase 5	0.1885	1.3393	0.000	0.000	86.6518	2.4752	1.4583	1.0516
Clase 6	0.9921	1.1359	0.000	0.0446	0.8681	88.3631	1.2748	0.9028
Clase 7	0.2679	2.2073	0.0050	0.3819	0.1438	0.0099	86.8899	2.9762
Clase 8	0.3472	7.8323	0.1835	0.0050	0.6994	1.8552	2.8423	77.6935

Tabla 4.18: Tabla de contingencia empleando normalización de 0 a 1 en imágenes con ruido para la SVM.

Se persigue el mismo objetivo que en la Sección 4.3.3 pero ahora para imágenes ruidosas, por ello, se procede a obtener el número óptimo de características de la misma manera empleando como método de extracción correspondiente, normalización de 0 a 1 y una resolución de 16×16 .

Número de características óptimo

Las Figuras 4.17, 4.18, 4.19 y 4.20 representan de forma gráfica cuál sería el número óptimo de características para dos casos concretos, el caso bueno (pareja 1) y el malo (pareja 2) mencionados anteriormente. Se puede observar como para la técnica de mínimos cuadrados la probabilidad de error es siempre 0 tanto para un caso (clase 1 frente a clase 3) como para el otro (clase 2 frente a clase 4).

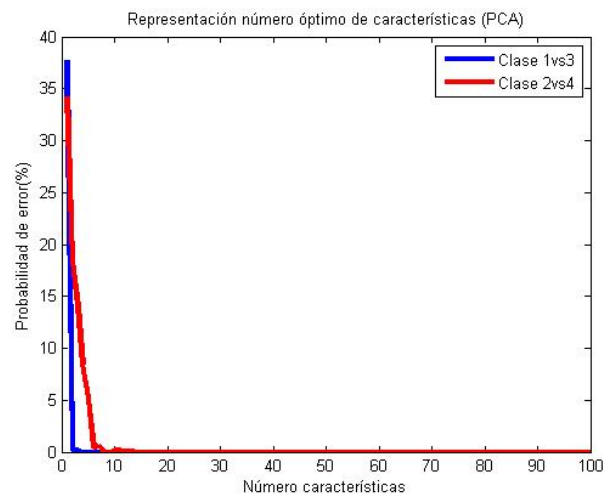


Figura 4.17: Representación de la probabilidad de error en función del número de características empleando como método de extracción PCA para la SVM.

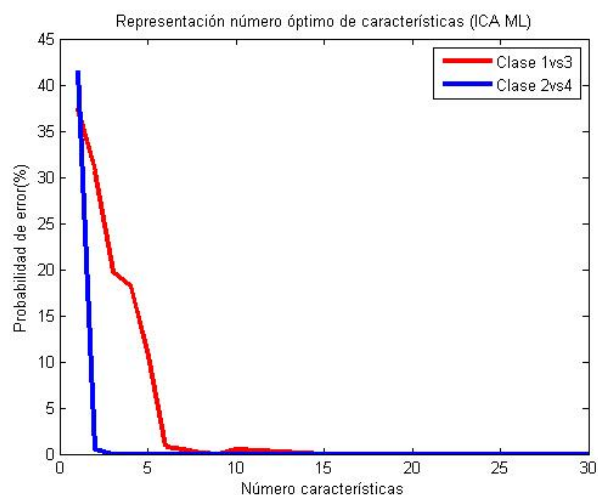


Figura 4.18: Representación de la probabilidad de error en función del número de características empleando como método de extracción ICA ML para la SVM.

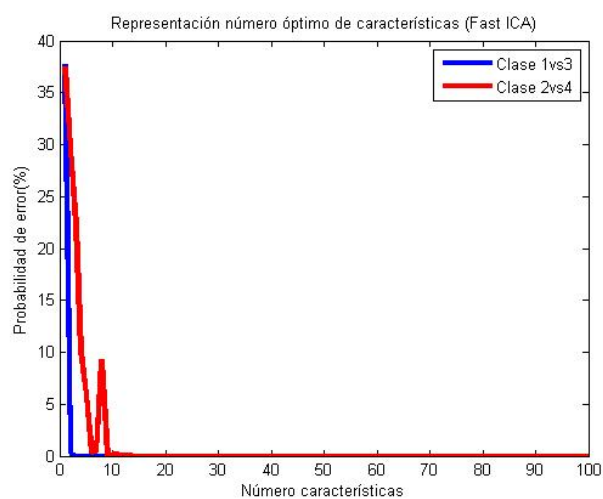


Figura 4.19: Representación de la probabilidad de error en función del número de características empleando como método de extracción Fast ICA para la SVM.

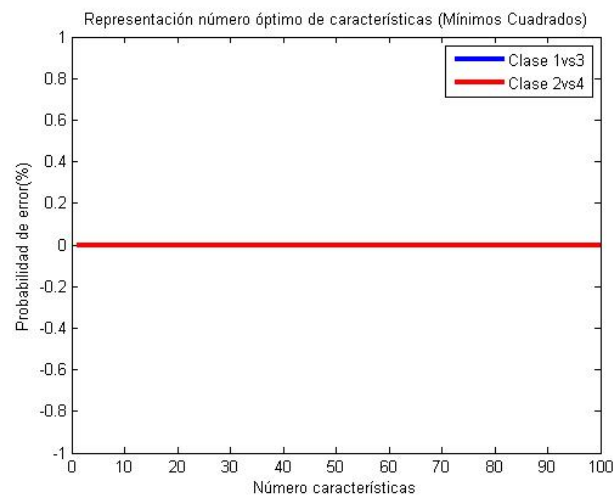


Figura 4.20: Representación de la probabilidad de error en función del número de características empleando como método de extracción Mínimos Cuadrados para la SVM.

Se proporcionan de nuevo dos tablas informativas, la Tabla 4.19 y la Tabla 4.20, que indican el número medio, mínimo y máximo de características empleadas. En concreto, la Tabla 4.19 muestra el número medio, mínimo y máximo de características en general requerido por cada método, mientras que la Tabla 4.20 muestra la misma información pero en este caso, por clases. En cuanto a técnicas no supervisadas consultando la Tabla 4.19, ICA ML es la que requiere un menor número medio de características. Sin embargo, la técnica supervisada LDA es en general la que menos características necesita. Por otro lado, si se consulta la Tabla 4.20, se puede comprobar que las clases más conflictivas y que requieren por tanto un mayor número de características son las clases 2, 4, 5, 7 y 8.

	PCA	ICA ML	Fast ICA	Mín. Cuadrados	LDA
Número medio	14.89	7.50	8.35	9.21	1
Mínimo	3	3	3	1	1
Máximo	80	20	25	80	1

Tabla 4.19: Tabla resumen del número características general requerido por cada método para la SVM.

Número de neuronas óptimo

En este momento se procede a calcular el número de neuronas óptimo para la GRBF. Se emplea como resolución de imagen 16x16 píxeles, normalización de 0 a 1 y el número de características óptimo obtenido para la SVM, y se calcula mediante validación cruzada el número óptimo de neuronas por cada par de clases.

Las Figuras 4.21, 4.22, 4.23 y 4.24 representan para el caso relativamente sencillo (caso bueno) y para el otro caso más complicado (caso malo) cuál sería el número óptimo de

		C 1	C 2	C 3	C 4	C 5	C 6	C 7	C 8
PCA	N.medio	5.71	20.28	6.71	12.57	24.57	10.42	15.85	23
	Mín.	3	8	4	3	4	4	4	3
	Máx.	15	50	9	50	80	25	50	80
ICA ML	N.medio	6	10.57	5.28	6.71	7	6.71	9	9.42
	Mín.	3	6	3	5	5	4	5	5
	Máx.	10	20	7	9	10	10	20	20
Fast ICA	N.medio	5.28	9.28	7.14	8.14	9.85	7.71	7.57	10.85
	Mín.	3	6	3	4	5	5	4	3
	Máx.	10	15	15	15	25	10	15	25
Min Cuadr	N.medio	8	1.14	5	19	15.71	12.28	11.42	1
	Mín.	1	1	1	1	1	1	1	1
	Máx.	50	2	25	50	80	80	50	1
LDA	N.medio	1	1	1	1	1	1	1	1
	Mín.	1	1	1	1	1	1	1	1
	Máx.	1	1	1	1	1	1	1	1

Tabla 4.20: Tabla resumen del número de características por clases requerido por cada método para la SVM.

neuronas en cada uno de los distintos métodos de extracción empleados. Se puede observar de nuevo el problema de los mínimos locales de Fast ICA.

Por otro lado, las Tablas 4.21 y 4.22 proporcionan información importante considerando cada una de las técnicas de extracción, a cerca del número medio, mínimo y máximo de neuronas obtenido tanto en general como por clases. Las técnicas que menor número de neuronas requieren son LDA e ICA ML que presenta un resultado prácticamente similar a Fast ICA. En este caso, las clases más conflictivas siguen siendo las clases 2, 4, 7 y 8.

	PCA	ICA ML	Fast ICA	Mín. Cuadrados	LDA
Número medio	7.78	6.53	6.82	3.25	1
Mínimo	2	2	2	1	1
Máximo	15	15	15	15	1

Tabla 4.21: Tabla resumen del número de neuronas general requerido por cada método para la GRBF.

Comparación de técnicas de extracción de características

En esta sección se presentan los resultados con imágenes ruidosas para una resolución de 16×16 píxeles empleando filtro de mediana.

La Tabla 4.23 representa los resultados obtenidos con cada una de las técnicas de extracción de características para SVM y GRBF. Como se puede observar en la tabla,

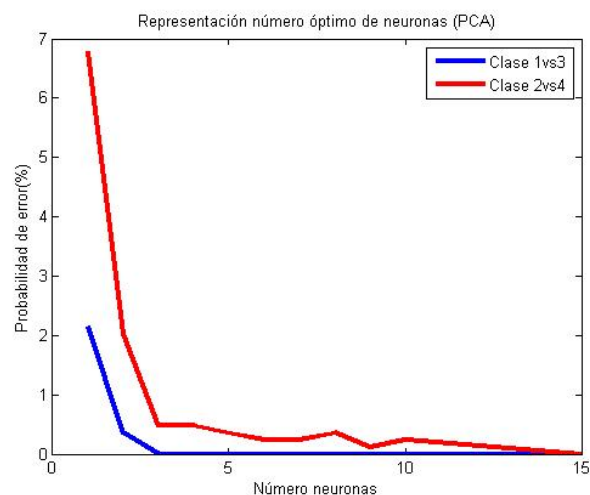


Figura 4.21: Representación de la probabilidad de error en función del número de neuronas empleando como método de extracción PCA para la GRBF.

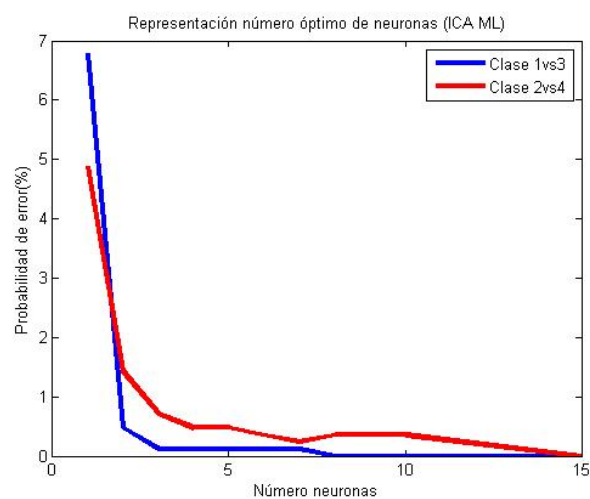


Figura 4.22: Representación de la probabilidad de error en función del número de neuronas empleando como método de extracción ICA ML para la GRBF.

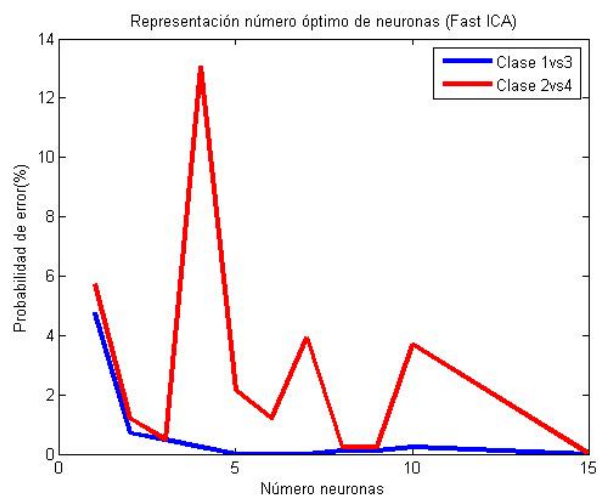


Figura 4.23: Representación de la probabilidad de error en función del número de neuronas empleando como método de extracción Fast ICA para la GRBF.

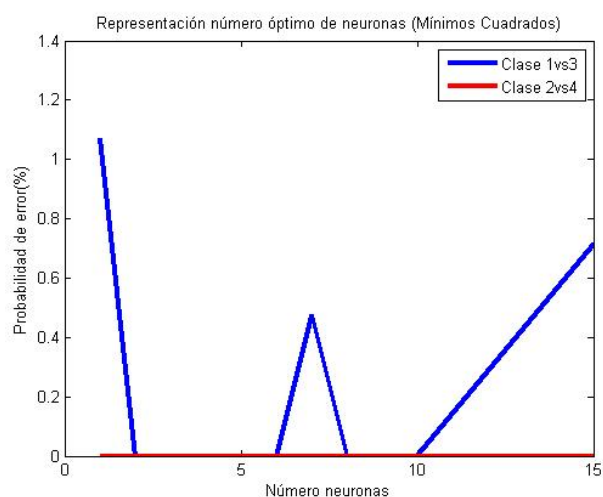


Figura 4.24: Representación de la probabilidad de error en función del número de neuronas empleando como método de extracción Mínimos Cuadrados para la GRBF.

		C 1	C 2	C 3	C 4	C 5	C 6	C 7	C 8
PCA	N.medio	8.85	7.85	5.28	7.85	6.14	6	10.71	9.57
	Mín.	3	2	2	3	2	2	3	2
	Máx.	15	15	15	15	15	9	15	15
ICA ML	N.medio	6.28	7.57	4.71	7.85	5.85	6.85	6.42	6.85
	Mín.	2	2	2	3	2	2	3	2
	Máx.	15	15	8	15	15	15	9	15
Fast ICA	N.medio	5.57	9.28	5.14	8.28	6.28	6.57	6.57	8.14
	Mín.	2	3	2	2	2	4	2	2
	Máx.	8	15	7	15	10	9	15	25
Min Cuadr	N.medio	4.14	3	2.14	4.42	2.14	2.57	2.57	4.14
	Mín.	2	1	1	1	1	1	1	3
	Máx.	15	9	5	15	4	5	6	5
LDA	N.medio	1	1	1	1	1	1	1	1
	Mín.	1	1	1	1	1	1	1	1
	Máx.	1	1	1	1	1	1	1	1

Tabla 4.22: Tabla resumen del número de neuronas por clases requerido por cada método para la GRBF.

para SVM la técnica no supervisada que mejor resultado presenta es ICA ML aunque el resto de técnicas no distan demasiado, mientras que para GRBF la que mejor funciona es PCA. Se puede comprobar que para ambos casos (SVM y GRBF) la técnica LDA presenta un resultado mayor con respecto a mínimos cuadrados, aunque la diferencia es mucho más significativa para SVM.

	PCA	ICA ML	Fast ICA	Min.Cuadrados	LDA
SVM	99.35	99.53	99.37	98.50	99.53
GRBF	99.11	98.99	99.02	98.48	98.80

Tabla 4.23: Probabilidades de acierto promedio obtenidas con cada una de las técnicas no supervisadas para SVM y GRBF.

La Tabla 4.24 muestra los resultados de ambas técnicas supervisadas empleando únicamente una característica y la Tabla 4.25 muestra los resultados de las técnicas supervisadas empleando una única neurona, en ambas tablas la técnica LDA presenta un resultado mejor que mínimos cuadrados.

Los resultados obtenidos con ruido para la resolución óptima (16×16 píxeles) siguen siendo mejores que los que obteníamos para las imágenes sin ruido con resolución 12×12 . Un motivo por el que puede estar sucediendo se puede deber a la resolución empleada, ya que, para imágenes con ruido es mayor y por tanto, puede estar proporcionando suficiente información como para producir mejores resultados. Sin embargo, como era de esperar según lo expuesto en la Sección 4.4, no es ese el motivo.

	Mín. Cuadrados	LDA
Prob. acierto promedio (%)	98.55	99.53

Tabla 4.24: Tabla comparativa de las técnicas supervisadas con una única característica para la SVM.

	Mín. Cuadrados	LDA
Prob. acierto promedio (%)	97.50	98.80

Tabla 4.25: Tabla comparativa de las técnicas supervisadas con una única neurona para la GRBF.

Cuando las imágenes empleadas presentan ruido hay que considerar además, que cada vez que se genera dicho ruido los resultados producidos en las imágenes son ligeramente distintos. El ruido no sigue la misma distribución y por tanto, los datos de entrenamiento y test son diferentes en cada simulación. Por este motivo, aunque la diferencia es muy pequeña lo correcto sería realizar una estimación empleando por ejemplo el método Montecarlo. Este método consiste en repetir un número de veces el experimento (100 por ejemplo) y obtener cuál es la desviación estadística del mismo. En este trabajo por restricciones de tiempo y recursos sólo se realiza una simulación, pero lo estricto sería llevar a cabo distintas simulaciones.

4.4.6. Técnicas de preprocesado

En cuanto a técnicas de preprocesado se ha estado empleando un filtro de mediana para tratar de combatir las irregularidades que pueden presentar las imágenes. Sin embargo, en este proyecto se han llevado a cabo también distintas pruebas empleando como técnica de preprocesado la detección de bordes. El objetivo es realizar un estudio comparativo para detectar en este caso qué es más efectivo, emplear un filtro de mediana o una detección de bordes. Tanto para SVM como para GRBF se han realizado las mismas pruebas (con ruido y sin él, con imágenes de resolución de 12×12 píxeles) y empleando las mismas técnicas de extracción de características, una técnica no supervisada (PCA) y otra supervisada (mínimos cuadrados). A continuación, la Tablas 4.26 y 4.27 representan los resultados obtenidos.

	SVM	GRBF
Prob. acierto promedio PCA (%)	95.42	94.08
Prob. acierto promedio Min.Cuadrados (%)	78.57	78.49

Tabla 4.26: Probabilidad de acierto promedio empleando la detección de bordes en imágenes sin ruido usando como técnicas de extracción PCA y mínimos cuadrados para la SVM y la GRBF.

Si se comparan los resultados con las mismas pruebas realizadas empleando filtro de mediana, se puede comprobar que el filtro de mediana proporciona mejores prestaciones.

	SVM	GRBF
Prob. acierto promedio PCA (%)	86.96	84.26
Prob. acierto promedio Min.Cuadrados (%)	60.57	61.09

Tabla 4.27: Probabilidad de acierto promedio empleando la detección de bordes en imágenes con ruido usando como técnicas de extracción PCA y mínimos cuadrados para la SVM y la GRBF.

Es destacable además que empleando la detección de bordes la técnica PCA funciona bastante mejor que la de mínimos cuadrados, que se ve mucho más afectada por el cambio.

4.4.7. SVM vs GRBF

Si se consideran los resultados proporcionados en la Sección 4.4.5, se puede comprobar que las prestaciones de la SVM son ligeramente superiores a los de la GRBF sea cual sea la técnica de extracción.

La Tabla 4.28 muestra de nuevo una comparación del número de parámetros necesarios en una SVM y una GRBF con las condiciones impuestas en la Sección 4.4.

	PCA	ICA ML	Fast ICA	Mín. Cuadrados	LDA
Núm. Parámetros SVM	131138	47207	67283	63895	8706
Núm. Parámetros GRBF	4892	2851	3213	3121	112

Tabla 4.28: Parámetros necesarios para SVM y GRBF con imágenes ruidosas.

Como se puede observar en la Tabla 4.28 la complejidad, es decir, el número de parámetros ha aumentado considerablemente con respecto a los mostrados en la Tabla 4.12 para imágenes sin ruido.

4.5. Bloque de extracción de características

Hasta ahora, para los resultados proporcionados en apartados anteriores la extracción de características se realizaba por parejas de clases empleando como método de clasificación todos contra todos. De esta forma, se obtenían unos parámetros de normalización, una matriz de proyección y unos parámetros de la máquina para cada pareja en particular. Por lo que el número de parejas que se podían formar con 8 clases era de 28. Cada clasificador por tanto, tenía unos parámetros específicos para cada pareja.

A continuación se estudia cuáles serían los resultados si la extracción de características en lugar de realizarse en parejas, se realizase de forma general. Es decir, cada clasificador tendría los mismos parámetros de normalización y la misma matriz de proyección, variando únicamente los parámetros de la máquina para cada pareja en cuestión.

4.5.1. Extracción de características en general SVM

Se emplea para ello una SVM con el método de extracción ICA ML y se prueban 3 valores, en concreto 5, 10 y 25 características. La Tabla 4.29 muestra cuáles son los resultados. Como se puede observar, a medida que aumenta el número de características van mejorando las prestaciones del sistema.

	Caract. 5	Caract. 10	Caract. 25
Prob. acierto promedio ICA ML (%)	84.85	99.48	99.95

Tabla 4.29: Probabilidad de acierto promedio realizando la extracción de características general para la SVM.

Se realizan las mismas pruebas empleando la extracción de características por parejas y se obtiene la Tabla 4.30.

	Caract. 5	Caract. 10	Caract. 25
Prob. acierto promedio ICA ML (%)	93.40	99.53	99.70

Tabla 4.30: Probabilidad de acierto promedio realizando la extracción de características por parejas para la SVM.

Hay que valorar qué es lo que compensa en cada momento. En el caso de la extracción general, se requiere un número de características más alto para alcanzar unas buenas prestaciones. Sin embargo, si se emplea la extracción de características por parejas, se consiguen buenos resultados con un número de características más bajo, pero a cambio aumenta la complejidad, aumenta la carga computacional. Para este caso en concreto, los resultados no distan demasiado.

A la vista de las Tablas 4.29 y 4.30, cuando se emplean 5 características, compensa realizar la extracción de características por parejas ya que, proporciona un resultado bastante bueno con respecto al obtenido en la extracción general. Sin embargo, cuando el número de características es mayor (10 ó 25 características), los resultados en ambos casos son prácticamente similares. Lo más conveniente en ese caso es emplear la extracción general, ya que, a igual número de características, es mucho más compleja la extracción de características por parejas.

4.5.2. Extracción de características en general GRBF

Se realizan las mismas pruebas para la GRBF empleando un número razonable de neuronas (en este caso, se han escogido 5 neuronas) y obteniendo como resultados los proporcionados en las Tablas 4.31 y 4.32.

La conclusión extraída en este caso es la misma que en la Sección 4.5.1. A la vista de las Tablas 4.31 y 4.32 con 10 características, la extracción general presenta un resultado favorable.

	Caract. 5	Caract. 10	Caract. 25
Prob. acierto promedio ICA ML (%)	77.42	97.25	99.85

Tabla 4.31: Probabilidad de acierto promedio realizando la extracción de características general para la GRBF.

	Caract. 5	Caract. 10	Caract. 25
Prob. acierto promedio ICA ML (%)	90.30	99.15	99.72

Tabla 4.32: Probabilidad de acierto promedio realizando la extracción de características por parejas para la GRBF.

Capítulo 5

Presupuesto y planificación del trabajo

En este capítulo se presenta por un lado la planificación del trabajo realizado y una estimación del presupuesto del trabajo realizado.

5.1. Planificación y seguimiento del trabajo

Llegados a este punto, es momento de exponer cómo se ha llevado a cabo la planificación del trabajo realizado y cuál ha sido la estrategia de seguimiento empleada.

El trabajo se ha llevado a cabo siguiendo una consecución de objetivos planificados con anterioridad por parte del tutor y de la alumna, siguiendo las pautas especificadas en la guía docente del Trabajo Fin de Grado. La organización del mismo se ha efectuado mediante revisiones semanales con el tutor donde se marcaban una serie de tareas a realizar. Las dudas, cuestiones o problemas puntuales han sido resueltos mediante correo electrónico independientemente de las reuniones pautadas.

En todo momento, se ha considerado la realización de un trabajo que se adaptase lo mejor posible a la temática de la especialidad cursada.

El *diagrama de Gantt* es una herramienta gráfica de planificación y programación de proyectos de diversa índole, cuyo objetivo es mostrar el tiempo de dedicación previsto para las diferentes tareas. Para ello, se va a emplear Microsoft Project, una herramienta que permite simplificar la planificación y administración de recursos, para abordar correctamente un proyecto. Antes de presentar el diagrama de Gantt se muestra la *red lógica Pert* que consiste en una representación gráfica de la sucesión de tareas en el tiempo (ver Figura 5.1). En él se puede visualizar el camino crítico (determinado por las cajas amarillas de la figura), es decir, aquel que determina la duración mínima de un proyecto. La siguiente Figura 5.2 muestra el diagrama de Gantt.

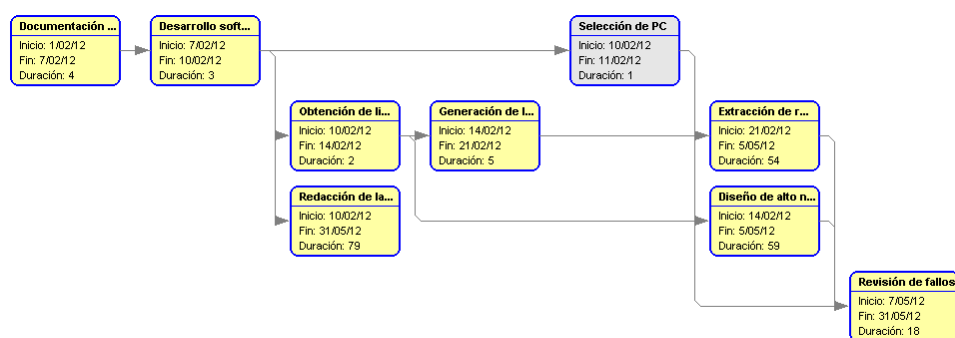


Figura 5.1: Red lógica Pert.

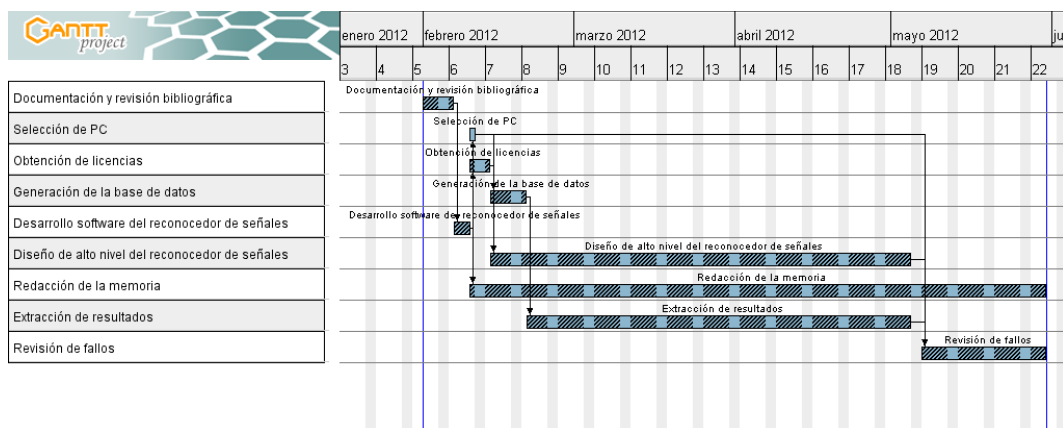


Figura 5.2: Diagrama de Gantt.

5.2. Presupuesto

A continuación se muestra el presupuesto con los costes del trabajo. La Figura 5.3 presenta un esquema básico de los recursos materiales y humanos empleados en este proyecto, así como el coste de cada uno de ellos. En cuanto a recursos humanos los sueldos se han calculado teniendo en cuenta la duración del proyecto (4 meses, comienza el 1 de febrero y concluye el 31 de mayo de 2012), el número de días (22 días sin considerar fines de semana) y horas trabajadas y el salario mínimo establecido por el Boletín Oficial del Estado (BOE). Según el BOE el sueldo estipulado para un licenciado es de 9,5 euros/hora, mientras que el de un diplomado es de 7,06 euros/hora. Teniendo en cuenta los datos proporcionados anteriormente se procede a calcular una estimación de los recursos humanos. Los recursos materiales y humanos forman parte de los costes directos¹, sin embargo, se incluye un 5 % de costes indirectos² provenientes de la amortización del equipo informático o de distintos trámites entre otros.

Recursos	Costes
Costes directos:	3.722,45 €
Recursos materiales:	933,33 €
PC	83,33 €
Licencias	250 €
Alquiler de local	600 €
Recursos Humanos:	2.789,12 €
Profesor	304 €
Estudiante en prácticas	2.485,12 €
Costes indirectos:	186,12€
TOTAL:	3.908,57€

Figura 5.3: Estimación presupuesto.

El coste del PC es de 1.000 euros en total, sin embargo, hay que tener en cuenta la amortización del mismo (amortización en el período del trabajo). Un equipo tiene una vida útil de 4 años aproximadamente, si se tiene en cuenta eso cada año es equivalente a un coste de 250 euros ($20,83 \text{ euros al mes} \times 4 \text{ meses} = 83,33 \text{ euros en 4 meses}$). También hay que amortizar el precio de las licencias (coste de 6.000 euros), activo intangible³ con una vida útil de 8 años, por lo que cada año supone un coste de 750 euros ($62,5 \text{ euros al mes} \times 4 \text{ meses} = 250 \text{ euros en 4 meses}$). Se requiere además el alquiler de un local para llevar a cabo la realización del trabajo, lo que supone un coste de 150 euros/mes (150

¹Costes directos, son aquellos costes imputables directamente a la realización de la actividad.

²Costes indirectos, son aquellos costes que dentro de un proyecto no son directamente imputables a una actividad sino al conjunto de ellas.

³El decreto 2649 de 1993 define los activos intangibles como aquellos recursos obtenidos por un ente económico que, careciendo de naturaleza material, implican un derecho o privilegio oponible a terceros, distinto de los derivados de los otros activos, de cuyo ejercicio o explotación pueden obtenerse beneficios económicos en varios períodos determinables, tales como patentes, marcas, derechos de autor, crédito mercantil, franquicias, así como los derechos derivados de bienes entregados en fiducia mercantil.

euros/mes \times 4 meses = 600 euros en 4 meses). Por otro lado, se procede a explicar cómo se ha realizado el cálculo del coste de los recursos humanos. El profesor supondrá un gasto de 304 euros, para ello, se consideran las horas trabajadas cada semana (2 h/semana), la duración del proyecto (4 meses) y el salario mínimo estipulado por hora (16 semanas \times 2 horas/semana \times 9,5 euros/hora = 304 euros en 4 meses). Para obtener los costes del trabajador en prácticas hay que tener en cuenta las mismas pautas (4 meses \times 22 días trabajados al mes \times 4 horas/día \times 7,06 euros/hora = 2.485,12), la única variación es el número de horas trabajadas durante el proyecto (4 horas/día).

Finalmente el coste total de llevar a cabo un proyecto de estas características, considerando todo lo expuesto anteriormente, es de 3.908,57 euros.

Capítulo 6

Conclusiones

En este Trabajo Fin de Grado se ha diseñado un clasificador para el reconocimiento de señales de tráfico que está formado por 4 módulos principales, el de preprocesado, normalización, extracción de características y el módulo máquina. En cada uno de ellos se han probado diferentes métodos para ver cuáles son sus prestaciones. Como técnicas de preprocesado se han utilizado el filtro de mediana y la detección de bordes. Se han probado varios tipos de normalización, de 0 a 1, de -1 a 1 y por varianza. Por otro lado, se han empleado las técnicas de extracción de características no supervisadas PCA, ICA-ML y Fast-ICA, y las técnicas supervisadas LS y LDA. Por último, en cuanto a métodos de aprendizaje máquina se han utilizado SVM y GRBF. Para evaluar los clasificadores propuestos, se ha generado una base de datos con imágenes de señales de tráfico.

En este capítulo se hablará de mejores prestaciones o de que un método funciona mejor que otro. En ese caso, nos referimos a que la probabilidad de acierto promedio es superior que a la que se obtiene con el resto de técnicas, es decir, el sistema comete menos fallos a la hora de clasificar a qué clase pertenecen cada una de las imágenes. A la vista de los resultados proporcionados en el Capítulo 4 se pueden extraer las siguientes conclusiones:

- Hay que elegir de forma correcta la base de datos con la que se va a trabajar. Esta debe representar efectos que puedan presentarse en una aplicación real como la considerada a escala en este trabajo.
- Es importante elegir bien los parámetros de la máquina para que no deriven en problemas de sobreajuste o subajuste. En nuestro caso al limitar el número de características y de neuronas a emplear, y al utilizar validación cruzada para la selección de la mejor máquina en la fase de entrenamiento se está controlando que no se produzcan situaciones de ese tipo.
- Como era de esperar, las técnicas de extracción de características supervisadas proporcionan mejores prestaciones que las no supervisadas con una complejidad más reducida (menor número de características).
- Para imágenes sin fuentes de ruido, las técnicas supervisadas presentan mejores prestaciones para SVM que para GRBF. Sin embargo, con las no supervisadas sucede al contrario, funciona mejor la red neuronal que la máquina de vectores soporte.

- A igualdad de condiciones empleando una única característica y una única neurona, si se comparan técnicas supervisadas, el método LDA proporciona prestaciones superiores al de mínimos cuadrados. En general, LDA presenta muy buenas prestaciones frente al resto de técnicas.
- Se puede decir que emplear como técnica de preprocesado un filtro de mediana mejora las prestaciones tanto para imágenes limpias como para imágenes con ruido sal y pimienta.
- Si se comparan las dos técnicas de preprocesado empleadas, filtro de mediana y detección de bordes, el filtro de mediana presenta elevadas prestaciones con respecto a las que se obtienen con detección de bordes.
- En este caso y para la base de datos empleada, el método de normalización que mejor funciona es la normalización de los datos de 0 a 1.
- En cuanto a técnicas no supervisadas, el método Fast ICA proporciona resultados ligeramente superiores al resto.
- La técnica no supervisada que menor número de parámetros requiere es ICA ML, mientras que como técnica supervisada LDA es la que menor memoria requiere con diferencia.
- Para imágenes con ruido, en este caso concreto, se necesita una resolución ligeramente superior (16×16 frente a 12×12 píxeles para imágenes sin ruido).
- Además, en imágenes ruidosas la SVM funciona siempre ligeramente mejor que la GRBF, aunque requiere una mayor complejidad.
- En imágenes ruidosas, las prestaciones obtenidas sin utilizar un filtro de mediana en la etapa de preprocesado no caen drásticamente. Esto puede deberse al filtrado implícito que supone bajar la resolución a 12×12 o 16×16 .
- Se ha comprobado que según aumenta la densidad de ruido en las imágenes, las prestaciones decaen ligeramente. Aunque hay que señalar que la degradación visual es mucho mayor que la disminución que sufren dichas prestaciones.
- Las clases más conflictivas y que requieren por tanto un mayor número de características y de neuronas, según los estudios realizados, son las clases 2, 4, 7 y 8.
- Es más conveniente a igualdad de prestaciones, realizar la extracción de características general, ya que la extracción por parejas es mucho más compleja computacionalmente, es decir, se obtiene un conjunto de parámetros únicos para cada par de clases (parámetros de normalización, matriz de proyección y parámetros de la máquina). Sin embargo, la extracción de características general conlleva una menor carga computacional, ya que los clasificadores comparten los mismos parámetros de normalización y la misma matriz de proyección, a excepción de los parámetros de la máquina (parámetros distintos para cada pareja de clases).

Finalmente, después de todo el análisis realizado se ha de elegir cuál es la solución propuesta para el clasificador. Para ello hay que tener en cuenta diferentes condiciones y ciertas restricciones. Por un lado, hay que considerar la calidad general de las imágenes a clasificar, considerando, por ejemplo, el nivel de ruido de las mismas. Para imágenes con alta calidad, como se ha visto en el análisis de imágenes sin ruido, utilizar la SVM con técnicas de extracción supervisadas, LDA, sería la mejor alternativa. En cambio, para imágenes con una degradación más notable, como se ha visto en los resultados presentados en el Capítulo 4 para imágenes ruidosas, la SVM proporciona mejores prestaciones que la GRBF. Por otro lado, hay que cotejar también lo restrictiva que sea la complejidad del clasificador, es decir, si existen restricciones sobre la complejidad del mismo para su implementación. En este caso, lo más conveniente es usar la GRBF ya que presenta menor coste computacional que la SVM. Sin embargo, si la complejidad del sistema es algo secundario, y con la tecnología empleada se puede implementar esta solución en tiempo real, es más apropiado emplear la SVM puesto que garantiza mejores prestaciones en esas condiciones. Por tanto, en general la decisión sobre qué propuesta es la que se va a llevar a cabo va a depender de las restricciones impuestas ante el sistema por el cliente y de las condiciones específicas de la aplicación.

Apéndice A

Glosario de acrónimos

- ATT: Transporte avanzado en telemática (*Advanced Transport Telematics*)
- BOE: Boletín Oficial del Estado
- BSS: Separación de fuentes independientes (*Blind Source Separation*)
- GPS: Sistema de posicionamiento global (*Global Positioning System*)
- GRBF: Función de base radial generalizada (*Generalised Radial Basis Function network*)
- IA: Inteligencia Artificial
- ICA: Análisis de componentes independientes (*Independent Component Analysis*)
- ITS: Sistema inteligente de transporte (*Intelligent Transport System*)
- KLT: Transformada de Karhunen-Loève (*Karhunen-Loève Transformation*)
- LDA: Análisis lineal discriminante (*Linear Discriminant Analysis*)
- LIBSVM: Librería SVM (*Library for Support Vector Machines*)
- LS: Mínimos cuadrados (*Least Square*)
- ML: Máxima verosimilitud (*Maximum Likelihood*)
- MLP: Perceptrón multicapa (*Multilayer Perceptron*)
- MI: Información mutua (*Mutual Information*)
- NG: No gaussianidad (*nongaussianity*)
- OLS: Mínimos cuadrados ortogonales (*Orthogonal Least Squares*)
- PCA: Análisis de componentes principales (*Principal Component Analysis*)
- RBF: Función de base radial (*Radial Basis Function network*)
- SVM: Máquina de vectores soporte (*Support Vector Machines*)

TICs: Tecnologías de la Información y las Comunicaciones

VANET: Red vehicular (*Vehicular Ad Hoc Network*)

V2I: Comunicación vehículo-infraestructura (*Vehicle-to-infrastructure communication*)

V2R: Comunicación vehículo-carretera (*Vehicle-to-roadside communication*)

V2V: Comunicación vehículo-vehículo (*Vehicle-to-vehicle communication*)

Lista de Figuras

1.1. Efecto visible de la contaminación en una gran ciudad, que en gran medida es producida por las emisiones de automóviles.	2
1.2. Vehículo del programa <i>Google Driverless Car</i> (fotografía de la agencia REUTERS) y conductor del mismo en una prueba de conducción en el estado de Nevada (You Tube).	4
1.3. Tres coches del fabricante Volvo siguen a un camión en un tren de carretera y siguen la misma ruta que este sin necesidad de conductor (fotografía de la agencia EFE publicada en el diario ABC).	4
1.4. Modo de funcionamiento multihop.	5
1.5. Esquema con el monitor del Opel Insignia para la monitorización del límite de velocidad (imagen superior) y fotografía con la proyección sobre el parabrisas de la información de limitación de velocidad, prohibición de adelantar, e información del navegador en un vehículo del fabricante BMW (imagen inferior). Ambas imágenes se han obtenido de la página del correspondiente fabricante.	6
2.1. Ejemplo de aumento de contraste en una imagen.	11
2.2. Presentación de imágenes con diferentes tipos de ruido.	11
2.3. Ejemplo de cambio de ejes PCA.	15
2.4. Arquitectura típica de una red neuronal hacia delante del Perceptrón Multicapa.	18
2.5. Ejemplo de la representación de muestras linealmente separables de dimensión 2 y distintas rectas que pueden separarlas, con sus correspondientes márgenes.	21
2.6. Hiperplano de separación óptimo para patrones linealmente separables.	22
2.7. Hiperplano de separación óptimo normalizado para patrones linealmente separables.	23
2.8. Problema no separable linealmente.	25
2.9. Proyección del espacio de entrada al espacio de características.	26
2.10. Función no convexa con múltiples mínimos locales.	27
2.11. Función convexa con un único mínimo local.	28

2.12. Problema sobreajuste.	29
2.13. Problema subajuste.	29
3.1. Representación del esquema básico del clasificador.	34
4.1. Imágenes que representan cada clase empleada.	38
4.2. Imágenes con distintas transformaciones.	39
4.3. Imágenes representativas de las clases 1 y 3 (Pareja 1 - Caso “bueno”). . . .	41
4.4. Imágenes representativas de las clases 2 y 4 (Pareja 2 - Caso “malo”). . . .	41
4.5. Representación de la probabilidad de error en función de la resolución para obtener el número óptimo de resolución para la SVM.	42
4.6. Representación de la probabilidad de error en función del número de ca- racterísticas empleando como método de extracción PCA para SVM. . . .	45
4.7. Representación de la probabilidad de error en función del número de ca- racterísticas empleando como método de extracción ICA ML para SVM. . .	45
4.8. Representación de la probabilidad de error en función del número de ca- racterísticas empleando como método de extracción Fast ICA para SVM. .	46
4.9. Representación de la probabilidad de error en función del número de carac- terísticas empleando como método de extracción Mínimos Cuadrados para SVM.	46
4.10. Representación de la probabilidad de error en función del número de neu- ronas empleando como método de extracción PCA para GRBF.	49
4.11. Representación de la probabilidad de error en función del número de neu- ronas empleando como método de extracción ICA ML para GRBF.	49
4.12. Representación de la probabilidad de error en función del número de neu- ronas empleando como método de extracción Fast ICA para GRBF.	50
4.13. Representación de la probabilidad de error en función del número de neuro- nas empleando como método de extracción Mínimos Cuadrados para GRBF. .	50
4.14. Degradación de imágenes según la densidad del ruido.	54
4.15. Representación de cómo afecta el filtro de mediana a imágenes con blurring. .	55
4.16. Representación de la probabilidad de error en función de la resolución para obtener el número óptimo de resolución para la SVM.	57
4.17. Representación de la probabilidad de error en función del número de ca- racterísticas empleando como método de extracción PCA para la SVM. . .	58
4.18. Representación de la probabilidad de error en función del número de ca- racterísticas empleando como método de extracción ICA ML para la SVM. .	59
4.19. Representación de la probabilidad de error en función del número de ca- racterísticas empleando como método de extracción Fast ICA para la SVM. .	59

4.20. Representación de la probabilidad de error en función del número de características empleando como método de extracción Mínimos Cuadrados para la SVM.	60
4.21. Representación de la probabilidad de error en función del número de neuronas empleando como método de extracción PCA para la GRBF.	62
4.22. Representación de la probabilidad de error en función del número de neuronas empleando como método de extracción ICA ML para la GRBF.	62
4.23. Representación de la probabilidad de error en función del número de neuronas empleando como método de extracción Fast ICA para la GRBF.	63
4.24. Representación de la probabilidad de error en función del número de neuronas empleando como método de extracción Mínimos Cuadrados para la GRBF.	63
5.1. Red lógica Pert.	70
5.2. Diagrama de Gantt.	70
5.3. Estimación presupuesto.	71

Lista de Tablas

4.1. Descripción de las señales de cada clase.	38
4.2. Probabilidades de error obtenidas para distintas resoluciones para SVM. . .	42
4.3. Tabla de contingencia empleando normalización de 0 a 1 para la SVM. . .	43
4.4. Tabla resumen con el número de características general requerido por cada método para la SVM.	45
4.5. Tabla resumen con el número de características por clases requerido por cada método para la SVM.	47
4.6. Tabla resumen con el número de neuronas general requerido por cada método para la GRBF.	48
4.7. Tabla resumen con el número de neuronas por clases requerido por cada método para la GRBF.	48
4.8. Probabilidades de acierto promedio obtenidas con cada técnica para SVM y GRBF empleando imágenes sin ruido.	51
4.9. Tabla comparativa de las técnicas supervisadas con una única característica para la SVM.	51
4.10. Tabla comparativa de las técnicas supervisadas con una única neurona para GRBF.	51
4.11. Probabilidades de acierto promedio obtenidas con cada técnica para SVM y GRBF en imágenes sin ruido utilizando un filtro de mediana en el pre-procesado.	52
4.12. Parámetros necesarios para SVM y GRBF con imágenes sin ruido sin aplicar filtro de mediana.	52
4.13. Probabilidades de acierto promedio obtenidas con cada técnica para SVM y GRBF empleando una densidad de ruido de 0.05.	53
4.14. Probabilidades de acierto para distintos valores de densidad de ruido para la SVM.	54
4.15. Probabilidades de acierto promedio obtenidas con cada técnica para SVM y GRBF en imágenes ruidosas utilizando un filtro de mediana.	56
4.16. Probabilidades de acierto promedio obtenidas con cada técnica para SVM y GRBF en imágenes ruidosas sin filtro de mediana.	56

4.17. Probabilidades de error obtenidas para distintas resoluciones con imágenes ruidosas para la SVM.	57
4.18. Tabla de contingencia empleando normalización de 0 a 1 en imágenes con ruido para la SVM.	58
4.19. Tabla resumen del número características general requerido por cada método para la SVM.	60
4.20. Tabla resumen del número de características por clases requerido por cada método para la SVM.	61
4.21. Tabla resumen del número de neuronas general requerido por cada método para la GRBF.	61
4.22. Tabla resumen del número de neuronas por clases requerido por cada método para la GRBF.	64
4.23. Probabilidades de acierto promedio obtenidas con cada una de las técnicas no supervisadas para SVM y GRBF.	64
4.24. Tabla comparativa de las técnicas supervisadas con una única característica para la SVM.	65
4.25. Tabla comparativa de las técnicas supervisadas con una única neurona para la GRBF.	65
4.26. Probabilidad de acierto promedio empleando la detección de bordes en imágenes sin ruido usando como técnicas de extracción PCA y mínimos cuadrados para la SVM y la GRBF.	65
4.27. Probabilidad de acierto promedio empleando la detección de bordes en imágenes con ruido usando como técnicas de extracción PCA y mínimos cuadrados para la SVM y la GRBF.	66
4.28. Parámetros necesarios para SVM y GRBF con imágenes ruidosas.	66
4.29. Probabilidad de acierto promedio realizando la extracción de características general para la SVM.	67
4.30. Probabilidad de acierto promedio realizando la extracción de características por parejas para la SVM.	67
4.31. Probabilidad de acierto promedio realizando la extracción de características general para la GRBF.	68
4.32. Probabilidad de acierto promedio realizando la extracción de características por parejas para la GRBF.	68

Bibliografía

- [Alvarez y Giraldo, 2008] D. A. ALVAREZ y E. GIRALDO. “ICA aplicado a la extracción de características en imágenes”. *Scientia et Technica*, XIV(40), diciembre 2008.
- [Arndt, 2008] M. ARNDT. “ITS in ETSI standards for road transport”. En *7th European Congress and Exhibition on Intelligent Transport Systems and Services*, Ginebra. Junio 2008.
- [Bakir et al., 2004] G. H. BAKIR, J. WESTON, y B. SCHOLKOPF. “Learning to find pre-images”. En *Advances in Neural Information Processing Systems (NIPS)*, volumen 16, págs. 449–456. 2004.
- [Bell y Sejnowski, 1995] A. BELL y T. J. SEJNOWSKI. “An information-maximization approach to blind separation and blind deconvolution”. *Neural Computation*, 7:1129–1159, 1995.
- [Bellman, 1961] R. BELLMAN. *Adaptive control processes : a guided tour*. Princeton University Press. 1961.
- [Bernardos et al., 2007] C. J. BERNARDOS, I. SOTO, M. CALDERÓN, F. BOAVIDA, y A. AZCORRA. “VARON: Vehicular ad-hoc route optimisation for NEMO”. *Elsevier Computer Communications*, 30(8):1765–1784, junio 2007.
- [Bishop, 2006] C. M. BISHOP. *Pattern recognition and machine learning*. Springer. 2006.
- [BMW España, 2012] BMW ESPAÑA. “BMW Connected Drive”. <http://www.bmw.es/connecteddrive>. 2012.
- [Bretscher, 1995] O. BRETSCHER. *Linear Algebra With Applications*, 3rd ed. Upper Saddle River NJ: Prentice Hall. 1995.
- [Broomhead y Lowe, 1988] D. S. BROOMHEAD y D. LOWE. “Multivariable functional interpolation and adaptive networks”. *Complex Systems Publications*, págs. 321–355, 1988.
- [Burges, 1998] C. J. BURGES. “A tutorial on support vector machines for pattern recognition”. *Data Mining and Knowledge Discovery*, (2):121–167, 1998.
- [Carlot, 2008] E. CARLOT. *Reconocimiento de patrones en imágenes médicas basado en sistemas inteligentes*. Tesis Doctoral, Universidad de Buenos Aires. 2008.

- [Chang y Lin, 2012] C.-C. CHANG y C.-J. LIN. “A library for support vector machines”. <http://www.csie.ntu.edu.tw/~cjlin/libsvm/>. 2012.
- [Chen et al., 1991] S. CHEN, C. F. N. COWAN, y P. M. GRANT. “Orthogonal least squares learning algorithm for radial basis function networks”. *IEEE Transactions on Neural Networks*, 2(2):302–309, marzo 1991.
- [Cisneros, 2009a] O. CISNEROS. “El sistema de detección de ángulos muertos y asistencia para cambio de carril”. Informe Técnico 40, Mecánica y electrónica. Instituto de Investigación sobre Reparación de Vehículos. 2009. e-Safety: Nuevas tecnologías al servicio de la seguridad vial.
- [Cisneros, 2009b] O. CISNEROS. “Los sistemas anticolidión (cas, collision avoidance systems)”. Informe Técnico 41, Mecánica y electrónica. Instituto de Investigación sobre Reparación de Vehículos. 2009. e-Safety: Nuevas tecnologías al servicio de la seguridad vial.
- [Cisneros, 2011] O. CISNEROS. “Nuevos sistemas de detección de peatones y frenado automático”. Informe Técnico 49, Mecánica y electrónica. Instituto de Investigación sobre Reparación de Vehículos. 2011. e-Safety: Nuevas tecnologías al servicio de la seguridad vial.
- [Collado et al., 2003] J. M. COLLADO, C. HILARIO, J. ARMINGOL, y A. DE LA ESCALERA. “Visión por computador para vehículos inteligentes”. *Ceautomática*, 2003.
- [Comisión Europea, 2007] COMISIÓN EUROPEA. “Resumen y publicación de mejores prácticas de seguridad vial en los estados miembros”. http://ec.europa.eu/transport/roadsafety_library/publications/supreme_c_es.pdf. 2007.
- [Delbracio y Mateu, 2006] M. DELBRACIO y M. MATEU. “Trabajo final de reconocimiento de patrones: Identificación utilizando PCA, ICA y LDA.”. https://iie.fing.edu.uy/investigacion/grupos/biometria/proyectos/patrones/RecPat_MM.pdf. 2006.
- [Diario ABC, 2011] DIARIO ABC. “El coche sin conductor, probado con éxito en Suecia”. <http://www.abc.es/20110119/tecnologia/abci-video-coche-conductor-probado-201101191025.html>. 2011. Publicado el 25 de enero de 2011.
- [Diario ABC, 2012a] DIARIO ABC. “El coche de BMW que se conduce solo”. <http://www.abc.es/20120124/ciencia/abci-coche-conduce-solo-201201240910.html>. 2012. Publicado el 24 de enero de 2012.
- [Diario ABC, 2012b] DIARIO ABC. “Volvo prueba en España su sistema de piloto automático para coches”. <http://www.abc.es/20120529/tecnologia/abci-coche-piloto-automatico-201205290811.html>. 2012. Publicado el 29 de mayo de 2012.
- [Diario Motor, 2008] DIARIO MOTOR. “Sistema de detección de señales y de salida de carril para el Insignia: Opel Eye”. <http://www.diariomotor.com>. 2008. Publicado el 19 de junio de 2008.
- [Diario Motor, 2011] DIARIO MOTOR. “Frenada automática de emergencia, obligatoria en camiones y autobuses a partir de 2013”. <http://www.diariomotor.com>. 2011. Publicado el 16 de mayo de 2011.

- [El Mundo, 2012] EL MUNDO. “Los vehículos sin conductor de Google obtienen licencia para circular en Nevada”. <http://www.elmundo.es/elmundo/2012/05/08/navegante/1336476861.html>. 2012.
- [Escalera, 2009] S. ESCALERA. “Symbols in natural scenes dataset”. <http://www.maia.ub.es/~sergio/>. 2009. Base de datos disponible en la sección “Projects” de la página personal de Sergio Escalera.
- [Flórez López y Fernández Fernández, 2008] R. FLÓREZ LÓPEZ y J. M. FERNÁNDEZ FERNÁNDEZ. *Las Redes Neuronales Artificiales. Fundamentos teóricos y aplicaciones prácticas*. Netbiblo. 2008.
- [Galaviz Mosqueda et al., 2011] G. A. GALAVIZ MOSQUEDA, R. AQUINO SANTOS, L. A. VILLASEÑOR GONZÁLEZ, A. EDWARDS BLOCK, y V. RANGEL LICEA. “Survey on multi-hop vehicular ad hoc networks under IEEE 802.16 technology”. *Intech*, Enero 2011.
- [Gonzalez y Woods, 2008] R. C. GONZALEZ y R. E. WOODS. *Digital Image Processing, 3rd edition*. Prentice Hall. 2008.
- [Gonzalez et al., 2009] R. C. GONZALEZ, R. E. WOODS, y S. L. EDDINS. *Digital Image Processing Using MATLAB, 2nd edition*. Gatesmark Publishing. 2009.
- [González Abril, 2003] L. GONZÁLEZ ABRIL. “Modelos de clasificación basados en máquinas de vectores soporte”. *Dialnet*, (17), 2003.
- [Hansen et al., 2001] L. K. HANSEN, J. LARSEN, y T. KOLENDA. “Blind detection of independent dynamic components”. En *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volumen 5, págs. 3197–3200, Salt Lake City, Utah, USA. 2001.
- [Hansen et al., 2006] L. K. HANSEN, J. LARSEN, y T. KOLENDA. “ICA:DTU Toolbox”. <http://cogsys.imm.dtu.dk/toolbox/ica/>. 2006. Versión 1.6, actualizada el 23 de junio de 2006.
- [Hartenstein y Laberteaux, 2010] H. HARTENSTEIN y K. P. LABERTEAUX. *VANET Vehicle Applications and Inter-Networking Technologies*. Wiley. 2010.
- [Haykin, 1994] S. HAYKIN. *Neural networks: A comprehensive foundation*. Macmillan Publishing Company. 1994.
- [Hsu y Lin, 2002] C. W. HSU y C. J. LIN. “A comparison of methods for multiclass support vector machines”. *IEEE Transactions of Neural Networks*, 13(2):415–425, 2002.
- [Hyvärinen et al., 2005] A. HYVÄRINEN, J. KARHUNEN, y E. OJA. “The FastICA package for MATLAB”. <http://research.ics.aalto.fi/ica/fastica/>. 2005. Versión 2.5, actualizada el 19 de noviembre de 2005.
- [Hyvärinen y Oja, 2000] A. HYVÄRINEN y E. OJA. “Independent component analysis: A tutorial”. Informe técnico, Helsinki University of Technology. 2000.

- [Instituto Español de Comercio Exterior, 2012] INSTITUTO ESPAÑOL DE COMERCIO EXTERIOR. “El sector de transporte en finlandia”. <http://www.icex.es/servicios/documentacion/documentoselaborados/icex/pdfs/nota%20transporte%20finlandia.pdf>. 2012.
- [Kwok y Tsang, 2004] J. T. KWOK y I. W. TSANG. “The pre-image problem in kernel methods”. *IEEE Transactions on Neural Networks*, 15(6):1517–1525, 2004.
- [Leiva Murillo, 2007] J. M. LEIVA MURILLO. *Extracción de características mediante criterios basados en teoría de la información*. Tesis Doctoral, Universidad Carlos III de Madrid. 2007.
- [Lázaro, 2001] M. LÁZARO. *Técnicas de modelado no lineal con incorporación de restricciones en las derivadas y su aplicación a comunicaciones*. Tesis Doctoral, Universidad de Cantabria. 2001.
- [Maldonado-Bascón et al., 2007] S. MALDONADO-BASCÓN, S. LAFUENTE-ARROYO, P. GIL-JIMÉNEZ, H. GÓMEZ-MORENO, y F. LÓPEZ-FERRERAS. “Road-sign detection and recognition based on support vector machines”. *IEEE Transactions on Intelligent Transportation Systems*, 8(2):264–278, junio 2007.
- [Marchena, 2000] C. S. C. MARCHENA. “Una aproximación a la teoría de los espacios de hilbert”. http://casanchi.com/casanchi_2000/23_hilbert01.pdf. 2000.
- [MathWorks, 2012] MATHWORKS. “Image processing toolbox”. <http://www.mathworks.es/products/image/>. 2012.
- [Merlo et al., 1997] G. MERLO, V. FERNÁNDEZ, F. CARAM, R. PRIEGUE, y R. GARCÍA MARTÍNEZ. “Reconocimiento de la voz mediante una red neuronal de Kohonen”. En *Congreso argentino de ciencias de la computación*, Universidad Nacional de la Plata, Buenos Aires. 1997.
- [Milone, 2005] D. H. MILONE. “Reconocimiento automático del habla con redes neuronales artificiales”. *Redalyc*, págs. 261–322, 2005.
- [Ministerio de la Presidencia, 2012] MINISTERIO DE LA PRESIDENCIA. “Real Decreto 662/2012 de 13 de abril, por el que se establece el marco para la implantación de los sistemas inteligentes de transporte (SIT) en el sector del transporte por carretera y para las interfaces con otros modos de transporte”. Boletín Oficial del Estado. 2012. Número de 14 de abril, Sección I, páginas 29524–29530.
- [Moody y Darken, 1990] J. E. MOODY y C. J. DARKEN. “Fast learning in networks of locally-tuned processing units”. *Proceedings of the IEEE*, 1(9):281–294, 1990.
- [Motsinger y Hubing, 2007] C. MOTSINGER y T. HUBING. “A review of vehicle-to-vehicle and vehicle-to-infrastructure initiatives”. Informe técnico, The Clemson University Vehicular Electronics Laboratory. 2007.
- [Navarrete et al., 2004] J. A. R. NAVARRETE, M. M. MADRID, E. B. QUEZADA, O. R. CANO, y J. M. F. ROMERO. “Aspectos de la fatiga del conductor y estudios de las tecnologías para detectarla y prevenirla”. Informe Técnico 241, Instituto Mexicano del Transporte. 2004.

- [Park et al., 2002] J.-W. PARK, R. HARLEY, y G. VENAYAGAMOORTHY. “Comparison of MLP and RBF neural networks using deviation signals for on-line identification of a synchronous generator”. En *IEEE Power Engineering Society Winter Meeting*, volumen 1, págs. 274–279. 2002.
- [Parlamento Europeo y Comisión, 2010] PARLAMENTO EUROPEO Y COMISIÓN. “Directiva 2010/40/UE del Parlamento Europeo y del Consejo de 7 de julio de 2010, por la que se establece el marco para la implantación de los sistemas de transporte inteligentes en el sector del transporte por carretera y para las interfaces con otros modos de transporte”. Diario Oficial de la Unión Europea. 2010. Número de 6 de agosto, páginas L207-1 a L207-13.
- [Poggio y Girosi, 1990] T. POGGIO y F. GIROSI. “Networks for approximation and learning”. *Proceedings of the IEEE*, 78(9):1481–1497, Septiembre 1990.
- [Powell, 1985] M. J. D. POWELL. “Radial basis functions for multivariable interpolation: A review”. En *IMA Conference on Algorithms for the Approximation of Functions and Data*, págs. 143–167. 1985.
- [Rodríguez García, 2009] M. A. RODRÍGUEZ GARCÍA. *Evaluación de Propuestas de Servicios de Localización para Redes VANET*. Tesis Doctoral, Facultad de Informática. Universidad de Murcia. 2009.
- [Rosenblatt, 1959] R. ROSENBLATT. *Principles of Neurodynamics*. Spartan Books, New York. 1959.
- [Santamaría et al., 1999] I. SANTAMARÍA, M. LÁZARO, C. J. PANTALEÓN, J. A. GARCÍA, A. TAZÓN, y ÁNGEL MEDIAVILLA. “A nonlinear MESFET model for intermodulation analysis using a generalized radial basis function network”. *Neurocomputing*, págs. 1–18, 1999.
- [Schölkopf et al., 1999] B. SCHÖLKOPF, C. J. C. BURGESS, y A. J. SMOLA. *Advances in Kernel Methods. Support Vector Learning*. Cambridge, Massachusetts. 1999.
- [Seijas, 2003] L. M. SEIJAS. *Reconocimiento de dígitos manuscritos mediante redes neuronales: una técnica híbrida*. Tesis Doctoral, Universidad de Buenos Aires. 2003.
- [Solera Ureña et al., 2006] R. SOLERA UREÑA, F. PÉREZ-CRUZ, y F. DÍAZ DE MARÍA. “Estimación de probabilidades a posteriori en SVMs multiclase de habla continua”. Informe técnico, Universidad de Zaragoza. IV Jornadas en Tecnología del Habla. 2006.
- [Unión Europea, 2012] UNIÓN EUROPEA. “Análisis de las tecnologías inalámbricas de comunicaciones (TIC) emergentes en el ámbito transfronterizo dentro del marco del proyecto I+E”. http://www.ipluse.eu/IMG/pdf/I_E__Poctefa_Estudio_A1_sectorTICs_Jun10.pdf, OPTnote="Unión Europea. 2012.
- [U.S. Department of Transportation, 2012] U.S. DEPARTMENT OF TRANSPORTATION. “Intelligent transportation systems - joint program office”. <http://www.its.dot.gov>. 2012.

- [van der Maaten, 2012] L. VAN DER MAATEN. “Matlab toolbox for dimensionality reduction”. http://homepage.tudelft.nl/19j49/Matlab_Toolbox_for_Dimensionality_Reduction.html. 2012.
- [Vapnik, 1999] V. N. VAPNIK. *The Nature of Statistical Learning Theory*. Springer. 1999.
- [Vicente et al., 2007] M. VICENTE, C. FERNÁNDEZ, A. GIL, y L. PAYÁ. “Equivalencia entre ICA y PCA como métodos de extracción de características en reconocimiento visual basado en apariencia”. *Ceautomática*, septiembre 2007.
- [Volvo Cars, 2012] VOLVO CARS. “Pedestrian detection with full auto brake - unique technology in the all-new Volvo S60”. <http://www.volvocars.com/en-ca/top/about/news-events/pages/default.aspx?itemid=17>. 2012.
- [Welling, 2005] M. WELLING. “Fisher linear discriminant analysis”. Informe técnico, Department of Computer Science. University of Toronto. 2005.
- [Weston y Watkins, 1998] J. WESTON y C. WATKINS. “Multi-class support vector machines”. Informe técnico, University of London. 1998.
- [Widrow, 1990] B. WIDROW. “30 years of adaptive neural networks: perceptron, madaline, and backpropagation”. *Proceedings of the IEEE*, 78(9):1415–1442, septiembre 1990.
- [Wikipedia, 2012] WIKIPEDIA. “Google’s driverless car”. http://en.wikipedia.org/wiki/Google_driverless_car. 2012.